SAPIENZA
UNIVERSITÀ DI ROMA

Federated Learning for 12-leads ECG arrhythmia classification: Preserving medical records privacy

Department of Computer, Control and Management Engineering (DIAG)

Master's degree (M.Sc.) in Data Science

Candidate

Daniel Mauricio Jiménez Gutiérrez

ID number 1939216

Thesis Advisor

Prof. Ioannis Chatzigiannakis

Co-Advisor

Prof. Andrea Vitaletti

Academic Year 2021/2022

Thesis defended on 18th of October 2022
in front of a Board of Examiners composed by:

Prof. Antonio Cianfrani (chairman)

Prof. Domenico Lembo

Prof. Fabio Galasso

Prof. Filomena Maggino

Federated Learning for 12-leads ECG arrhythmia classification: Preserving medical records privacy

Master's thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: jimenezgutierrez.1939216@studenti.uniroma1.it

Dedicated to God, my family and my love one

Acknowledgments

I want to convey my gratitude to professors Ioannis Chatzigiannakis, my supervisor, and Andrea Vitaletti, my co-advisor, for guiding me through this project with endeavour and commitment. It was a pleasant and educational journey to work under his leadership. I also extend this acknowledgement to Lorella and Hassan, my colleagues in this thesis, since they made spectacular contributions to this project.

I want to express my gratitude to my girlfriend Sonia, who supported me while developing this thesis with love and patience.

And last but not least, I would also like to thank family and friends for their support and contributions while studying in a foreign country.

Contents

1 Introduction **1**
1.1 Document's outline . . . 2

2 Cardiological fundamentals **3**
2.1 The human heart nature . . . 3
2.2 ECG's history . . . 4
2.3 The ECG Waveform . . . 6
2.4 The 12-leads ECG . . . 7
2.5 Arrhythmias Types . . . 8
2.5.1 Sinus . . . 9
2.5.2 Atrial . . . 12
2.5.3 Junctional . . . 15
2.5.4 Ventricular . . . 17
2.5.5 AV Blocks . . . 20

3 Related work **23**
3.1 Bibliographical research methodology . . . 23
3.2 ECG classification . . . 27
3.2.1 Techniques to handle imbalanced data . . . 28
3.2.2 Methods for ECG classification . . . 28
3.2.3 Metrics for ECG classification . . . 29
3.3 Federated learning for ECG . . . 30
3.3.1 Methods for ECG classification using Federated Learning . . . 30
3.3.2 Methods to handle NON-IID data . . . 32
3.3.3 Metrics for Federated Learning . . . 36

4 Analytical techniques and tools **41**
4.1 Commonly used techniques and tools . . . 41
4.1.1 Data Wrangling (DW) . . . 41
4.1.2 Feature Engineering (FE) . . . 42
4.1.3 Exploratory Data Analysis (EDA) . . . 44
4.1.4 Unbalanced classes . . . 45
4.1.5 Machine Learning Models . . . 46
4.1.6 Metrics . . . 50
4.2 Federated learning (FL) fundamentals . . . 52
4.2.1 Definition . . . 52

4.2.2 FL types 53
4.2.3 Advantages and disadvantages 57
4.2.4 Proposed approaches 58

5 ECG Arrhythmias classification **63**
5.1 Dataset employed 63
5.2 Centralized Learning 67
5.2.1 Data wrangling 67
5.2.2 EDA 67
5.2.3 Feature Selection and normalization 70
5.2.4 Balancing classes (arrhythmias) 72
5.2.5 Fitted models and results 73
5.3 Federated Learning 75
5.3.1 IID approach 75
5.3.2 Non-IID approach 79

6 Conclusions **85**
6.1 Summary 85
6.2 Future Developments 85

Bibliography **87**

Chapter 1

Introduction

The Fourth Industrial Revolution is fostering the emergence of new scenarios in which vast volumes of data are shared among independent and potentially disparate organizations. Often used on a cross-border basis to improve shared services in several sectors, such as finance and health care. Despite its benefits, technological advancements are introducing new security and privacy concerns associated with the use of these data, which include factors such as collection, analysis, usage, storage, and sharing. Indeed, in the case of personal information, incorrect usage, unsafe storage, data leakage, or misuse can all compromise a person's privacy. As a result, when personal data is subject to federated computation, the availability and proper use of privacy-preserving and fairness-aware mechanisms are presented as a key element to be addressed to increase people's trust and thus achieving the sustainable and ethical realization of these scenarios.

Digital Health Products (DHP) in the eHealth sector, in particular, present unique options to provide efficient, effective, cross-border high-quality healthcare services [36]. Today, cutting-edge AI-based medical data analysis has promise for early detection, faster diagnosis, better decision-making, and more successful treatment, according to [1]. The use of AI-based DHP in healthcare operations, services, and applications has created a significant and pressing need to combine highly private medical data gathered from a variety of sources. It also includes millions of parameters that must be learned from sufficiently big, curated datasets to reach clinical-grade accuracy while remaining safe, fair, and equitable, as well as generalizing well to previously unseen [58].

Federated learning (FL) is an architecture that aims to solve the problem of data governance and privacy by collectively training algorithms without transferring data. It was originally designed for a variety of domains, including mobile and edge device use cases, but it has recently acquired popularity in healthcare applications [46]. FL allows for collaborative insights, such as in the form of a consensus model, without transferring patient data outside of the institutions' firewalls. Instead, each

participating institution's ML process takes place locally, with only model weights being shared. Models trained by FL can achieve performance levels comparable to those trained on centrally hosted data sets and superior to models that only see isolated single-institutional data, according to recent studies [46].

1.1 Document's outline

The present thesis document is organized into six chapters.

The first chapter (1) introduces the reasons behind the need for a machine learning solution to classify the 12-leads, encouraging the introduction of Federated Learning in the architecture. The second chapter (2) describes the essential background to understand the cardiological basics. It contains the human heart nature and all the concepts associated with ECG monitoring, including the most common arrhythmia types.

The third chapter (3) focuses on the state-of-the-art regarding the research topic. It contains information for both ECG classification and Federated Learning. In the fourth chapter (4), the analytical methodologies and tools are introduced and explained in detail.

The fifth chapter (5) it is exposed the results of analyzing the proposed data under both the Centralized and Federated Learning environment. Finally, the last chapter (6) is dedicated to the overall conclusions achieved by the research and to possible future developments.

Chapter 2

Cardiological fundamentals

[21] "An electrocardiogram (ECG) is a measure of how the electrical activity of the heart changes over time, as action potentials within each myocyte propagate throughout the heart as a whole during each cardiac cycle. In other words, the ECG is the recording of the cumulative signals produced by populations of cells eliciting changes in their membrane potentials at a given point in time. The ECG provides specific waveforms of electrical differences when the atria and ventricles depolarize and repolarize."

2.1 The human heart nature

For the purposes of an ECG, the human body can be thought of as a large volume conductor. It is made up of tissues and a conductive media in which the heart is suspended. The heart contracts during the cardiac cycle in response to coordinated action potentials traveling through the chambers of the heart. One section of the heart tissue is depolarized, while another is at rest or polarized, as is usual.

The intensity of the voltages observed is determined by the electrodes' orientation in relation to the dipole ends. The signal amplitudes are proportional to the mass of tissue used to create the dipole at any particular time. Electrodes are typically placed on the skin's surface to detect the voltages of these electrical fields, giving rise to the ECG [21].

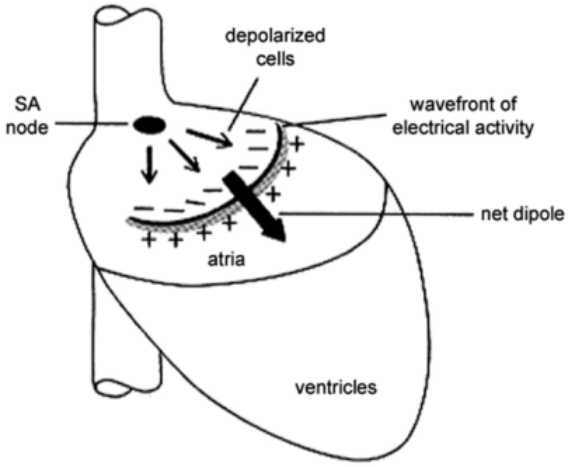


Figure 2.1. After conduction begins at the sinoatrial node, cells in the atria begin to depolarize. This creates an electrical wavefront that moves down toward the ventricles, with polarized cells at the front. The separation of charge results in a dipole across the heart (the large black arrow shows its direction) [21].

2.2 ECG's history

The discovery of intrinsic electrical activity within the heart dates all the way back to the 1840s. Carlo Matteucci, an Italian physicist, was the first to discover that each heartbeat is accompanied by an electrical current in 1842. Emil DuBois-Reymond, a German scientist, published the first action potential associated with muscular contraction not long after. In 1856, Rudolph von Koelliker and Heinrich Miller used a galvanometer to record the first cardiac action potential. Following that, Augustus D. Waller recorded the first human ECG after Gabriel Lippmann invented the capillary electrometer in the early 1870s. That first device is shown in Figure 2.2.

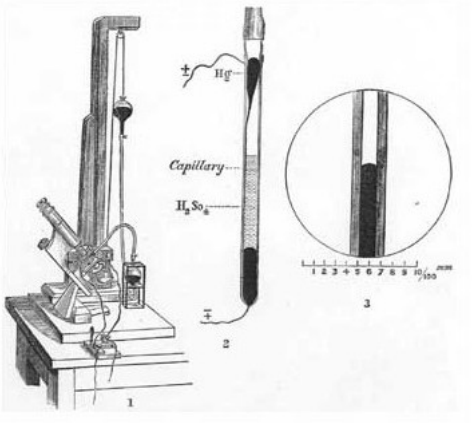


Figure 2.2. Lippmann electrometer

Willem Einthoven's creation of the string galvanometer in 1901 was a key milestone in cardiac electrocardiography. The next year, he published the first ECG using his string galvanometer. Einthoven's string galvanometer consisted of a huge electromagnet with a thin silver-coated string stretched across it; electric currents passing through the thread caused the string to move from side to side in the electromagnet's magnetic field.

Einthoven made yet another significant addition to cardiac electrophysiology in 1912, when he discovered a mathematical link between the direction and size of the deflections recorded by the three limb leads. Einthoven's triangle is the name for this hypothesis. Before Frank Wilson described unipolar leads and the precordial lead configuration, the typical three-limb leads were used for three decades. The traditional Einthoven limb leads, as well as the precordial and unipolar limb leads based on Wilson's work, make up the 12-lead ECG layout now in use.

This instrument was initially manufactured in 1905 by the Cambridge Scientific Instrument Company in London. Electrical impulses were sent from a hospital over a mile away to Einthoven's laboratory via a telephone cable. Bedside machines, on the other hand, were not available until the 1920s. The Sanborn Company produced a smaller version of the unit in 1935 that weighed only about 25 pounds.

Figure 2.3. Holter-Edan ECG device

With Norman Jeff Holter's invention of the Holter monitor in 1949, the use of ECG in a nonclinical context became viable. The first iteration of this device was a 75-pound backpack that could record the ECG continually and send the signals via radio. The size of subsequent iterations of such devices has been drastically decreased, and the signal is now recorded digitally. Miniaturized devices now allow

patients to be monitored for longer periods of time (typically 24 hours) to aid in the diagnosis of any rhythm or ischemic heart disease concerns. One of the latest versions of the ECG is the one appearing in Figure 2.3.

2.3 The ECG Waveform

Signals of voltage versus time are created during the recording of an ECG, which are generally shown in millivolts (mV) vs seconds. Figure 2.4 depicts a typical Lead II ECG waveform. The negative electrode was placed on the right wrist and the positive electrode on the left ankle for this Lead II ECG recording. As a result, a series of peaks and waves can be seen, each of which corresponds to ventricular or atrial depolarization and repolarization, with each segment of the signal indicating a separate event in the cardiac cycle.

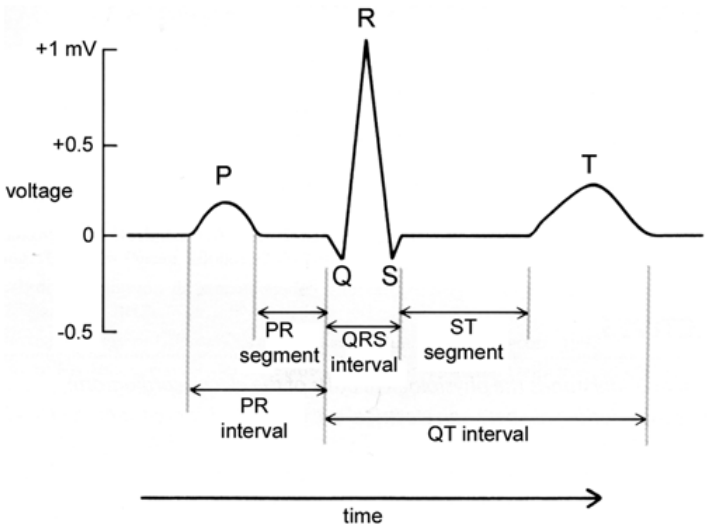


Figure 2.4. A typical ECG waveform for one cardiac cycle, measured from the Lead II position [21].

Three principal waveforms are recorded by the ECG (2.4):

- The P-wave
- QRS complex
- T-wave.

The P-wave is created by depolarisation of the atria, the QRS by depolarisation of the ventricles, and the T-wave by repolarisation of the ventricles. In most people, these waveforms occur in a repeating rhythm called sinus rhythm, so called because

it originates in the sinus node. In some people, a fourth waveform (not shown in the previous image) called a U-wave can be seen. This is usually seen at slower heart rates. The significance of the U-wave remains uncertain. Some authors think that it represents the late stages of ventricular repolarisation, while others describe it as a post-repolarisation phenomenon. U-wave abnormalities have been described in various disease states including ischaemic heart disease [48].

The depolarization of the sinoatrial node, which is positioned within the right atrium, starts the typical cardiac cycle. A conventional ECG will not detect this early firing because the node does not have enough cells to provide a measurable electrical potential. The right and left ventricles continue to depolarize after the P wave, resulting in the recordable QRS complex, which lasts about 100 milliseconds. The Q-wave is the initial negative deflection (if present), the R-wave is the largest positive deflection, and the S-wave is the smallest positive deflection [49].

The T-wave is usually the last potential in a cardiac cycle, followed by the P-wave of the next cycle, and so on. The ECG signal returns to baseline near the conclusion of ventricular contraction, and the ventricles repolarize after contraction. Atrial contractions have stopped and the atria are repolarizing at the same time as the QRS complex. Because the effects of this widespread atrial repolarization are obscured by the much larger volume of tissue engaged in ventricular depolarization, it is not generally detectable in an ECG [21].

2.4 The 12-leads ECG

An ECG lead is a recording of the heart's electrical activity as seen from one side. As a result, when we take a 12-lead ECG, we're recording cardiac electrical activity from 12 different angles [49]. Assume you're visiting a historic structure and taking images of it. If you snap 12 photos from different angles around the structure, each one will depict a distinct element, such as the front, sides, and back. They work together to provide a three-dimensional record of the structure's shape and appearance. In a similar way, a 12-lead ECG creates a three-dimensional depiction of the heart's electrical activity.

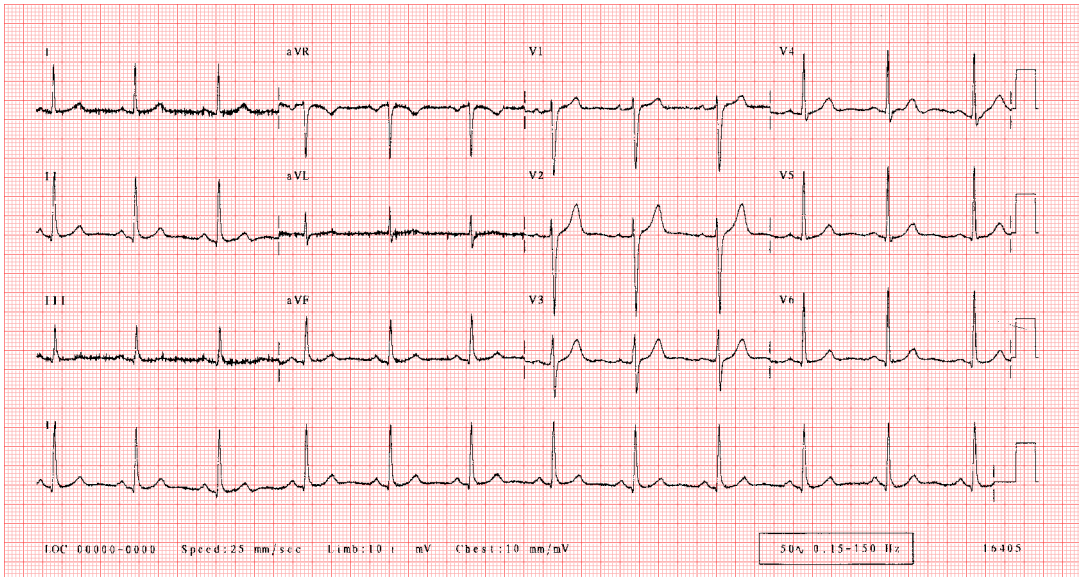


Figure 2.5. 12-leads normal ECG

Multiple images of the heart's electrical activity can be recorded depending on the type of machine utilized and the number of electrodes inserted. The usage of 12-lead ECG devices is common among health-care professionals. Twelve separate electrical images of the heart are measured and recorded via a 12-lead ECG (Figure 2.5). In other words, it records the electrical activity of the heart as observed from 12 various angles. For example, Lead II monitors electrical activity as observed from the heart's inferior (diaphragmatic) surface. This lead is frequently used to measure heart rate [57].

2.5 Arrhythmias Types

Analyzing arrhythmias is a difficult undertaking since every person on the planet has a unique ECG that differs from everyone else's, and one person's ECG can change dramatically from one second to the next. Memorizing some of the most common ECG patterns and attempting to recognize them in the future is insufficient. Pattern identification is a popular yet unintentional way of approaching arrhythmias via ECG analysis ([16]). Often the Arrhythmias are divided into two global categories:

1. **Rhythmic**: Determined as a sequence of uneven beats
2. **Morphological**: Made of abnormal single beat

The work presented in the current thesis is focused on first type of classification. Those arrhythmias have a categorization provided by SNOMED CT. The latter

is the world's most complete and precise terminology package, with widespread acceptance around the world. It provides a common language for clinical IT systems, making data exchange between them easier, safer, and more accurate. It covers everything from processes and symptoms to clinical measurements, diagnosis, and drugs, and it's all in one place.

For the Physionet 2020 challenge were only considered around 27 arrhythmias, which are the most frequently found regarding the ECG analysis. Nevertheless, along the following paragraphs there is a deep explanation of all the possible arrhythmias. To see the complete list of categories it is possible to read the GitHub cited in [37].

Arrhythmias are often divided into groups based on where the rhythm is initiated by the pacemaker. The following are the most prevalent sites, and consequently the primary arrhythmia categories:

1. Sinus
2. Atrial
3. Junctional
4. Ventricular
5. AV Blocks

2.5.1 Sinus

It is necessary to comprehend the 'benchmark' rhythm, or hemodynamically perfect rhythm, which is referred to as **Normal Sinus Rhythm** and sometimes abbreviated to **NSR**, in order to assess cardiac rhythms (Figure 2.6). The following features must be present in order for a rhythm to be classified as Normal Sinus Rhythm:

Characteristic	**Status**
Rhythm	Regular
Rate	60-100/minute
p waves	Present, upright, symmetrical, one before every QRS
pri	.12-.20 seconds
QRS	.06-.10 seconds

Table 2.1. Characteristics of Normal Sinus Rhythm

Figure 2.6. Normal Sinus Rhythm ([16])

Sinus Bradycardia (SB): When a patient's heart rate falls below 60 beats per minute, they are said to be bradycardic. Slow heart rates can be seen in fit and active people who are usually asymptomatic. When a patient's heart rate falls below 60 beats per minute, critical care nurses must be ready to assess for decreasing cardiac output right away.

Characteristic	**Status**
Rhythm	Regular
Rate	< 60/minute
p waves	Present, upright, symmetrical, one before every QRS
pri	.12-.20 seconds
QRS	.06-.10 seconds

Table 2.2. Characteristics of Sinus Bradycardia

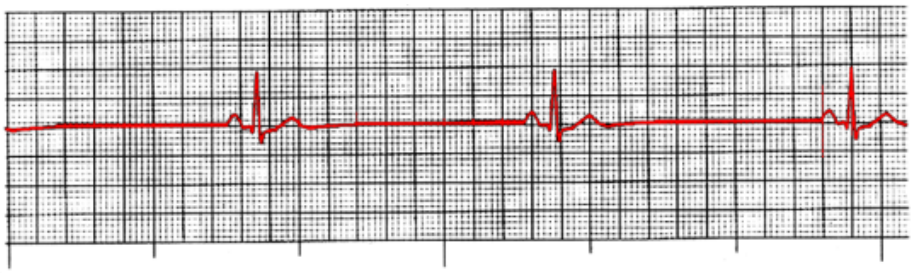


Figure 2.7. Sinus Bradycardia ([16])

Sinus Tachycardia (STach): When a patient's heart rate exceeds 100 beats per minute, they are labeled tachycardic, though most people don't notice symptoms until their heart rate exceeds 150 beats per minute. At this point, a critical care nurse should look for signs and symptoms of decreased cardiac output (such as hypotension or a loss of consciousness).

Characteristic	Status
Rhythm	Regular
Rate	>100/minute
p waves	Present, upright, symmetrical, one before every QRS
pri	.12-.20 seconds
QRS	.06-.10 seconds

Table 2.3. Characteristics of Sinus Tachycardia

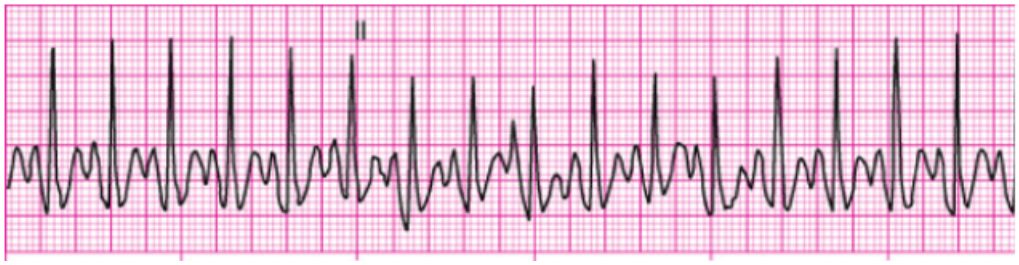


Figure 2.8. Sinus Tachycardia ([16])

Sinus Arrhythmia (SA): This arrhythmia is typically benign and do not require any sort of treatment. It is seen in children and also in mechanically ventilated patients.

Characteristic	Status
Rhythm	Regular
Rate	60-100/minute
p waves	Present, upright, symmetrical, one before every QRS
pri	.12-.20 seconds
QRS	.06-.10 seconds

Table 2.4. Characteristics of Sinus Arrhythmia

Figure 2.9. Sinus Arrhythmia ([16])

Wandering Atrial Pacemaker (WAP): It can be a normal aberration associated to Ischemia. There is no treatment required.

Characteristic	Status
Rhythm	Regular
Rate	60-100/minute
p waves	P waves vary in shape and size
pri	.12-.20 seconds
QRS	.06-.10 seconds

Table 2.5. Characteristics of Wandering Atrial Pacemaker

Figure 2.10. Wandering Atrial Pacemaker ([16])

2.5.2 Atrial

The rhythms that originate in the atrial will be examined in the following section. Premature atrial contractions, atrial flutter, atrial fibrillation, and supraventricular tachycardia are examples of these arrhythmias. The key characteristics of cardiac rhythms will be outlined, as well as nursing consequences and useful advice to help critical care nurses correctly interpret atrial arrhythmias.

Premature Atrial Contractions (PAC): it can be a normal aberration, Ischemia, or a signal of atrial irritability. It can lead to more serious atrial rhythms.

Characteristic	Status
Rhythm	Early beat (PAC) causes rhythm to be irregular
Rate	Underlying rhythm usually 60-100/minute
p waves	P waves have different configuration than underlying rhythm
pri	.12-.20 seconds in underlying rhythm
QRS	.06-.10 seconds in underlying rhythm

Table 2.6. Characteristics of Premature Atrial Contractions

Figure 2.11. Premature Atrial Contractions ([16])

Atrial Flutter (AFL): It is caused by electrolyte imbalance, Hypertension, Ischaemic heart disease, Congenital heart disease, Rheumatic valve disease. Also after a cardiac surgery.

Characteristic	Status
Rhythm	Regular or irregular
Rate	60-100/minute (ventricular rate) 250-400 (atrial rate)
p waves	No p waves present. Flutter waves (F waves) or 'sawtooth' waves
pri	No pri since no p waves
QRS	.06-.10 seconds

Table 2.7. Characteristics of Atrial Flutter

Figure 2.12. Atrial Flutter ([16])

Atrial Fibrillation (AF): It is caused by Electrolyte imbalance, Hypertension, Ischaemic heart disease, Congenital heart disease, Rheumatic valve disease. Also following a cardiac surgery.

Characteristic	Status
Rhythm	Irregular
Rate	60-100/minute (ventricular rate) >400/minute (atrial rate)
p waves	No p waves. Fibrillatory waves (f waves)
pri	No No pri since no p waves
QRS	.06-.10 seconds

Table 2.8. Characteristics of Atrial Fibrillation

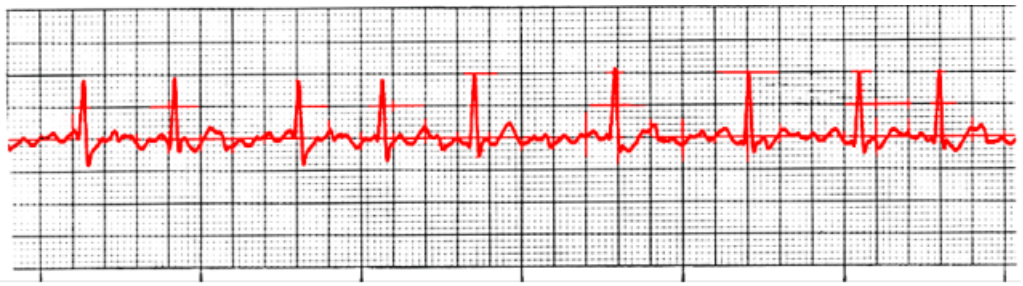


Figure 2.13. Atrial Fibrillation ([16])

Supraventricular Tachycardia (SVT): It is caused by Congenital, heart disease, Emotional stress, Physical stress or exertion, Illegal drugs (i.e. Cocaine or ecstasy), Alcohol, Caffeine.

Characteristic	Status
Rhythm	Regular
Rate	150-250/minute (atrial rate)
p waves	P waves may not be seen at higher rates
pri	.12-.20 seconds (if seen)
QRS	.06-.10 seconds

Table 2.9. Characteristics of Supraventricular Tachycardia

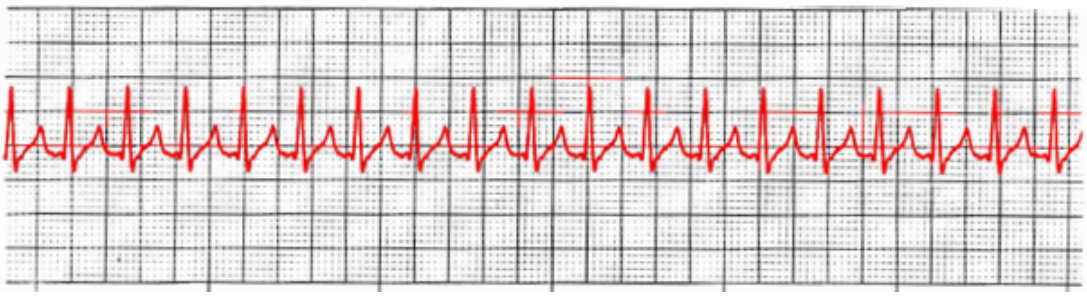


Figure 2.14. Supraventricular Tachycardia ([16])

2.5.3 Junctional

Junctional rhythms are temporary and non-lethal rhythms that originate in the AV node or junctional region. Inverted p waves are a typical feature of all junctional rhythms. Premature junctional contractions, junctional rhythm, and paroxysmal junctional tachycardia are among the rhythms covered in this section ([16]).

Premature Junctional Contraction (JPC): It is caused by Medication toxicity (i.e. digoxin), Ischemia. There is not treatment required. Continue to observe for increasing number of JPCs since this indicates increasing AV node irritability.

Characteristic	Status
Rhythm	Early beat (PJC) causes the rhythm to be irregular
Rate	60-100/minute (underlying rhythm)
p waves	P waves inverted or not seen in JPC
pri	Not applicable
QRS	.06-.10 seconds (in underlying rhythm)

Table 2.10. Characteristics of Premature Junctional Contraction

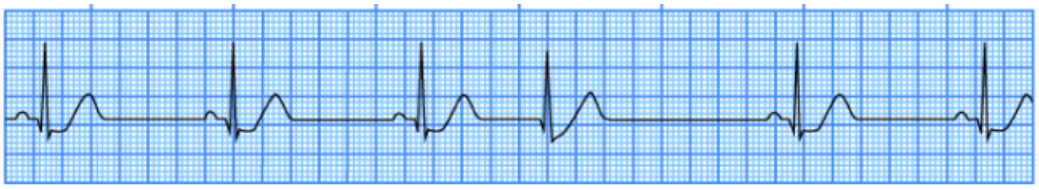


Figure 2.15. Premature Junctional Contraction ([16])

Junctional Rhythm (AVJR): It is caused by Medication toxicity (i.e. digoxin) or ischemia. It is necessary to treat causes.

Characteristic	Status
Rhythm	Regular
Rate	<60/minute
p waves	P waves inverted or absent
pri	.12-.20 seconds
QRS	.06-.10 seconds

Table 2.11. Characteristics of Junctional Rhythm

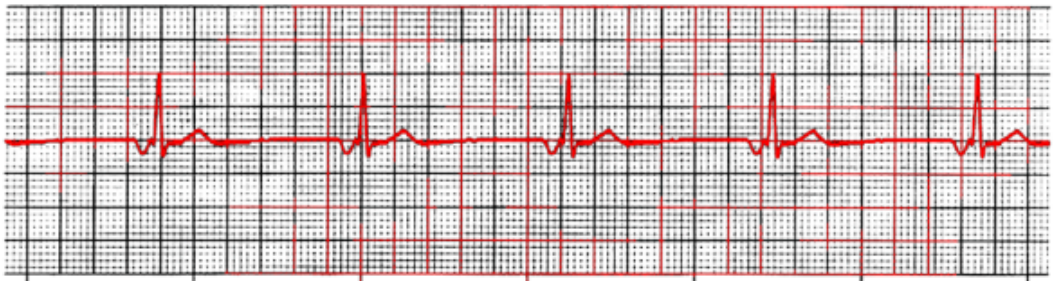


Figure 2.16. Junctional Rhythm ([16])

Accelerated Junctional Rhythm (AJR): It is caused by Medication toxicity (i.e. digoxin) or ischemia. It is necessary to treat causes.

Characteristic	Status
Rhythm	Regular
Rate	60-100/minute
p waves	P waves inverted or absent
pri	Not applicable
QRS	.06-.10 seconds

Table 2.12. Characteristics of Accelerated Junctional Rhythm

Figure 2.17. Accelerated Junctional Rhythm ([16])

Paroxysmal Junctional Tachycardia (JT): It is caused by ischemia. Its treatment is the same than SVT.

Characteristic	Status
Rhythm	Regular
Rate	150-250/minute
p waves	P waves inverted or absent (if seen)
pri	Not applicable
QRS	.06-.10 seconds

Table 2.13. Characteristics of Paroxysmal Junctional Tachycardia

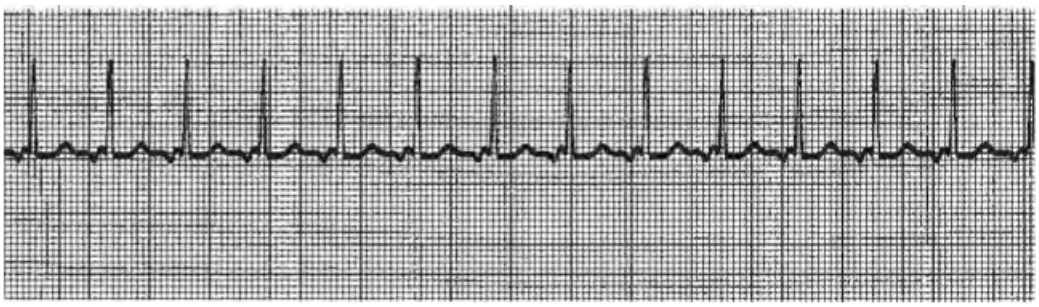


Figure 2.18. Paroxysmal Junctional Tachycardia ([16])

2.5.4 Ventricular

Premature ventricular contractions, ventricular tachycardia, and ventricular fibrillation are examples of ventricular rhythms that can be induced by irritability, as well as those that result from the failure of higher-level pacemakers. Irritability patients have significantly various treatment options and consequences.

Premature Ventricular Contractions (PVC): It is caused by Ventricular irritability (i.e.hypoxemia, acid-base imbalance, medications, electrolyte imbalance).

Characteristic	Status
Rhythm	Early beat (PVC) causes the rhythm to be irregular
Rate	60-100/minute (underlying rhythm)
p waves	None (in PVC)
pri	None (in PVC)
QRS	> .12 seconds (wide and bizzare)

Table 2.14. Characteristics of Premature Ventricular Contractions

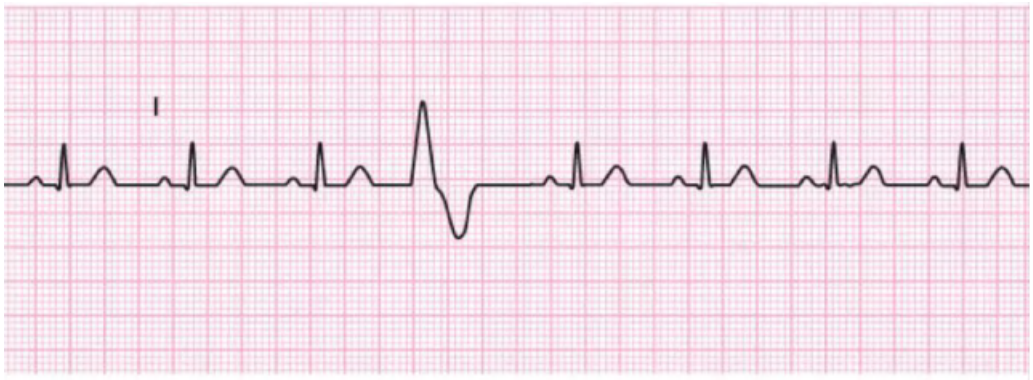


Figure 2.19. Premature Ventricular Contractions ([16])

Ventricular Tachycardia (VTach): It is caused by Ventricular irritability (i.e.hypoxemia, acid-base imbalance, medications, electrolyte imbalance).

Characteristic	Status
Rhythm	Regular
Rate	150-250/min
p waves	None
pri	None
QRS	> .12 seconds (wide and bizzare)

Table 2.15. Characteristics of Premature Ventricular Contractions

Figure 2.20. Premature Ventricular Contractions ([16])

Ventricular Fibrillation (VF): It is caused by Ventricular irritability (i.e.hypoxemia, acid-base imbalance, medications, electrolyte imbalance).

Characteristic	Status
Rhythm	Irregular and chaotic
Rate	Cannot calculate
p waves	None
pri	None
QRS	None

Table 2.16. Characteristics of Ventricular Fibrillation

Figure 2.21. Ventricular Fibrillation ([16])

Idioventricular Rhythm (IR): It is caused by Ischemia, reperfusion post thrombolytics.

Characteristic	Status
Rhythm	Regular
Rate	<40/minute
p waves	No p waves
pri	No pri
QRS	> .12 seconds (wide and bizarre)

Table 2.17. Characteristics of Idioventricular Rhythm

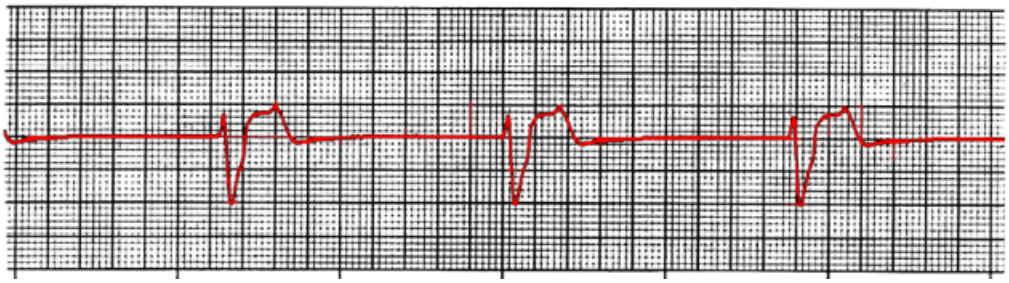


Figure 2.22. Idioventricular Rhythm

2.5.5 AV Blocks

Electrical conduction failure via the myocardium is characterized by atrioventricular (AV) blockages. Because AV blockages are linked to severe risk worsening or haemodynamic impairment, the critical care nurse must recognize and treat them as soon as possible. 1st degree heart block, 2nd degree heart block (Mobitz type 1 or Wenkebach), and 3rd degree heart block are all types of AV block (complete heart block). ([16])

First Degree AV Block (IAVB): It is caused by AV nodal disease, Enhanced vagal tone (i.e. athletes), Myocarditis, Following Myocardial Infarction, Electrolyte disturbances, Medications (i.e. Calcium channel blockers, Beta blockers).

Characteristic	Status
Rhythm	Regular
Rate	60-100/minute
p waves	P waves normal
pri	>.20 seconds
QRS	.06-.10 seconds

Table 2.18. Characteristics of First Degree AV Block

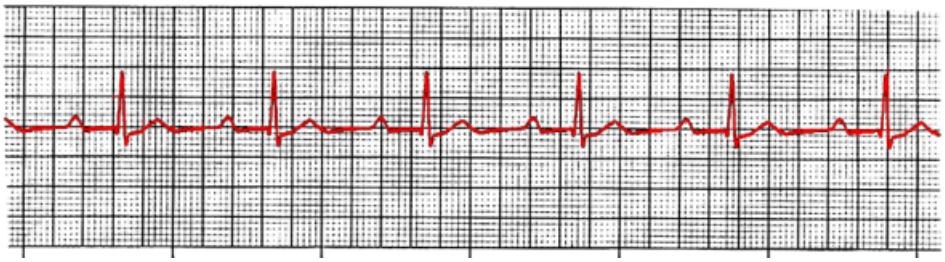


Figure 2.23. First Degree AV Block ([16])

Second Degree Type I (IIAVB): It is caused by Ischemia. Usually benign,

with no treatment required. If patient becomes haemodynamically compromised interventions for bradycardia should be considered.

Characteristic	Status
Rhythm	Regular or slightly irregular
Rate	60-100/minute
p waves	P waves normal
pri	Progressively gets longer until a beat is dropped
QRS	.06-.10 seconds

Table 2.19. Characteristics of Second Degree Type I

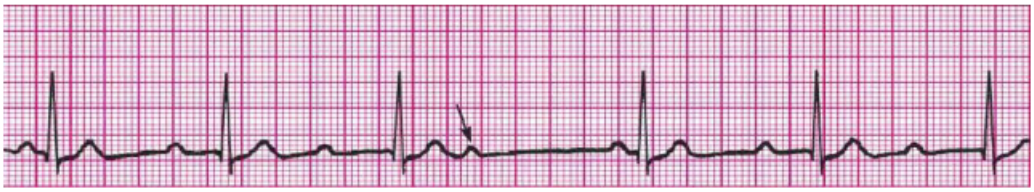


Figure 2.24. Second Degree Type I ([16])

Chapter 3

Related work

3.1 Bibliographical research methodology

To define the state-of-the-art for ECG classification and Federated learning I performed a reduced Systematic Review. The latter is defined as [17] a 'process of critically evaluating, summarizing, and seeking to reconcile the evidence.' In other words, it is a complete evaluation of literature that differs from a traditional review in that it is undertaken in a methodical (or systematic) manner, following a pre-specified process to avoid bias, with the goal of synthesizing the information gathered.

Then the first step to perform the systematic review (SR or bibliographical research) was to decide the reference and citation databases to use. In the table 3.1 are written the academic search engines used to retrieve the related documents.

Search Engine	Definition	Link
Google Scholar	A free web search engine that indexes the full text or metadata of scholarly literature from a variety of publishers and fields.	Link to GS
PubMed.gov	Contains almost 34 million citations from MEDLINE, life science journals, and online books for biomedical literature.	Link to PM
IEEEXplore	A research database that allows users to access journal articles, conference proceedings, technical standards, etc. in computer science, electrical engineering, and electronics.	Link to IE
Scopus	Has a huge collection of Physical Sciences and Engineering papers, from foundational science to novel and unique research, and spanning many disciplines both theoretical and applied.	Link to SCs
Web Of Science	It is the most reliable publisher-independent worldwide citation database in the world.	Link to WoS
Papers with code	Their goal is to provide a free and open library that includes Machine Learning articles, code, datasets, methodologies, and evaluation tables.	Link to PwC

Table 3.1. Databases (search engines) used to find documents

Each one of the search engines showed in the previous table work based on a "query" which will contain the information of the topic desired. Besides, to get better results, it is recommended to use multiple queries that may enrich the matter in research. Those queries are listed below, grouped by the specific topic to be found about:

1. **Federated learning Generalities**: To retrieve the conceptual definition and approximations of Federated Learning I used the following queries:
    - Federated learning arrhythmia
    - Federated learning ECG
    - Federated learning healthcare arrhythmia
    - Federated learning healthcare iot
    - Federated learning IoT ecg
    - Federated learning healthcare low power mobile
    - Federated learning PhysioNet
    - Federated learning TensorFlow Lite
2. **Options for ECG classification**: To get the different methods used along the history in the ECG classification theme I employed the next queries:

- Machine Learning ECG arrhythmia
- Deep Learning ECG arrhythmia
- Machine Learning IoT ECG arrhythmia
- Deep Learning IoT ECG arrhythmia

3. **Non-IID Methods**: To evaluate the different methods to deal with Non Independent Nor Identical distributed (Non-IID) data, I search over the following queries:

   - Federated learning non iid
   - Federated learning independent identically distributed

4. **Imbalanced data**: To check the techniques used b other authors regarding the class (labels, response variable) imbalanced, I used the next queries:

   - Imbalanced data ecg
   - Imbalanced data federated learning

5. **Metrics**: To gather the most used metrics in both ECG and Federated Learning ECG classifications I employed the following queries:

   - Federated learning metrics
   - ECG classification metric

6. **Federated learning types**: To understand the possible architectures of Federated Learning I was based on the next queries:

   - Federated learning architecture
   - Types federated learning

7. **Arrhythmia types**: To know the different arrhythmia classifications, I employed the following query:

   - Cardiac Arrhythmia types

To gather in a proper and technical way all the documents and paers found I employed the PRISMA Flow. As cited in [26], the latter is a flow diagram to depict the flow of studies through the different phases of the systematic review. That tool widely used for reporting original systematic reviews. Thus, the PRISMA flow employed is depicted in Figure 3.1.

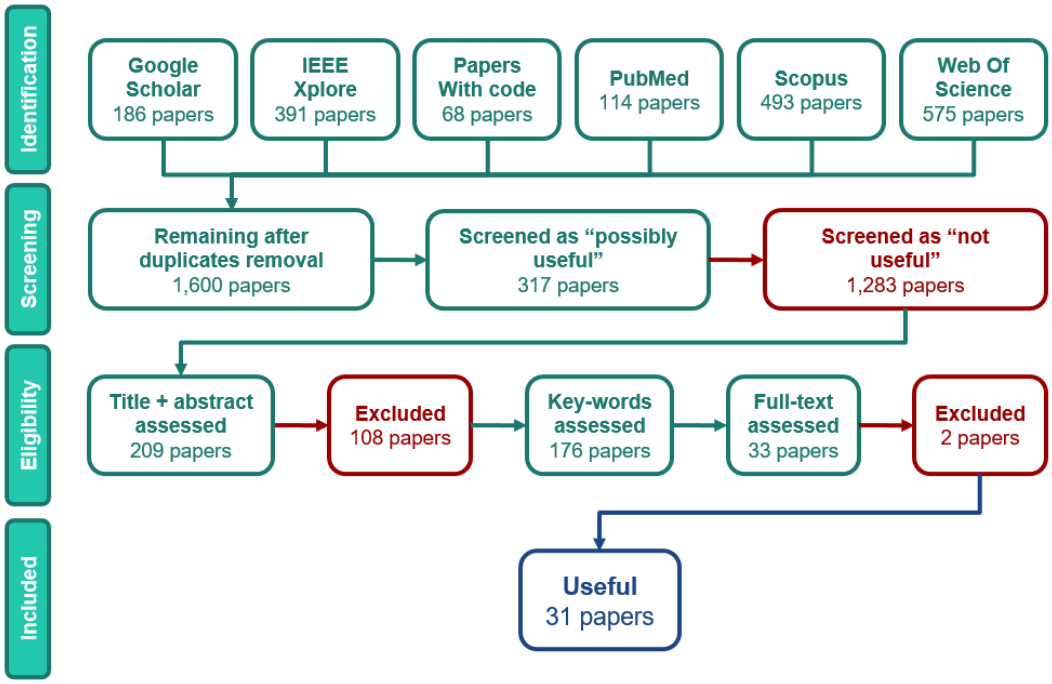


Figure 3.1. Number of papers retrieved by year of publication and database

As shown in figure 3.1, along the phase of Identification, 1,827 articles where found using the queries mentioned previously. It is important to mention that for each one of those queries, I downloaded the results retrieved in BibTex and CSV formats, depending on the search engine used. With the files saved it was possible to retrieve some generalities from the documents. As an example, in figure 3.2 it is depicted the number of papers found in each search database divided by publication year.

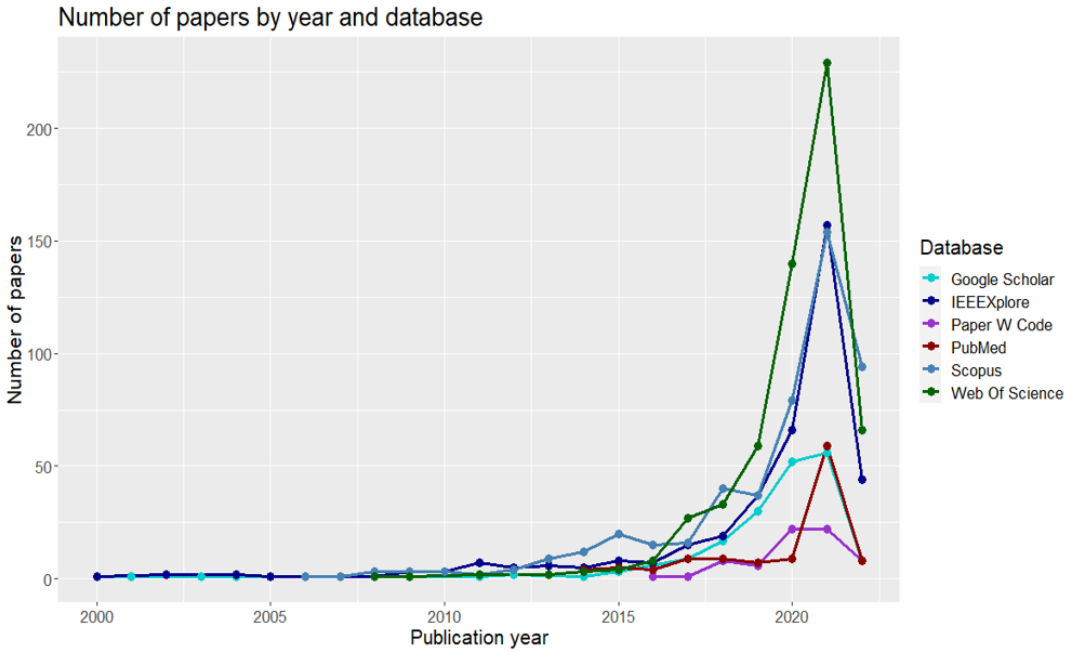


Figure 3.2. PRISMA flow for gathering documents

From the previous graph we can see that most of the papers where published after 2017. That is because the concept of Federated Learning was introduced by Google in that year. Nevertheless, ECG classification is a topics that has been worked back in the 2000s. Moreover, the search engine that produced the highest number of resources was Web Of Science.

Continuing with the analysis of figure 3.1, during the Screening phase I removed the duplicated documents keeping 1,600. After a fast screening (only title), I ended up with 317 possibly useful resources. During the eligibility phase, I screened the latter (title and abstract), ending up with 209 promising papers. Next, after assessing the keywords (given in the queries), I reached 176 papers that were not read but were used to extract those keywords. On the other hand, I read 33 documents , from where 2 where not useful. That's the process how I found the 31 most useful papers after the research.

3.2 ECG classification

After reading the aforementioned documents, I gather some relevant highlights regarding the classification of ECG signals. Along the following paragraphs I summarize the most important findings at each topic.

3.2.1 Techniques to handle imbalanced data

In general, imbalanced data describes datasets in which the target class has an unequal distribution of observations. For example, when one class label has a large number of observations while the other has a small number.The authors who have dealt with this topic followed different paths to tackle this issue. Authors from [9] introduced a *Balanced Accuracy (BACC)* and the *Matthew's Correlation Coefficient (MCC)* to correct the fact that the classes don't share a similar distribution. In the article [43] it was introduced the *Generative Adversarial Network (GAN)* which deals with imbalanced data by generating and using additional fake data for detection purpose. In addition, [32] used the *Synthetic Minority Oversampling Technique (SMOTE)*, which is an oversampling technique.

Other approaches also include the so-called *Ratio Loss* ([59]) where the global node estimates the composition data each round. When detecting an imbalanced composition continuously, the system acknowledges the class imbalance and load the Ratio Loss. One final possibility is the *Recall of data* in which one randomly augment the lower class and in each training epoch change the selected individuals. Then, there are enough possibilities tried in the literature, each of of them with their pros and cons that can be verified further.

3.2.2 Methods for ECG classification

Along the literature I could find that a huge amount of diverse techniques have been applied when classifying ECG' arrhythmias. As an example [51], [56], and [33] focused their efforts on Using Deep Neural Networks (Artificial Neural Networks and Multi-layer Perceptron) to get a model that predicts the abnormality given the ECG signal. In comparison, the author of [50] combined in his paper the use of Naïve Bayes, Adaboost, Random Forest and Support Vector Machines to get the best classifier for his paper. Finally, [10], [18] and [5] employed in their research some Convolutional Neural Network approaches. Among them the highlighted Squeezenet, Attention mechanism and Resnet as the champion methods to deal with the ECG detection.

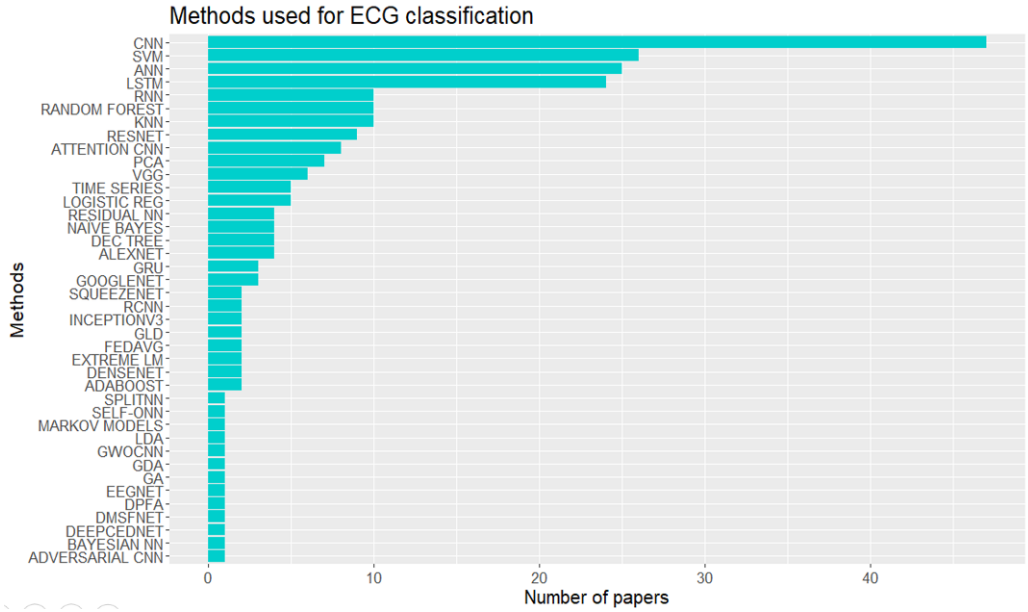


Figure 3.3. Most used classification methods for ECG

As depicted in figure 3.3, the most used technique is the Convolutional Neural Network (CNN). That one includes also some self-made Deep Neural Networks (DNN). On the second place we find Support Vector Machines (SVM) and Artificial Neural Networks (ANN). An very close to them most of the author also used Long-Short Term Memory (LSTM) algorithms. On the opposite, a few papers contributed with techniques like GWOCNN, DFPA, DEEPCETNET, etc, which are also CNN but that have specific alterations adapted to by the papers' authors.

3.2.3 Metrics for ECG classification

With respect to ECG arrhythmia classification, there are plenty of measurements employed in the literature. In figure 3.4 I show the most relevant metrics used in this aim, gathered from the available articles and papers shown in chapter 3.1.

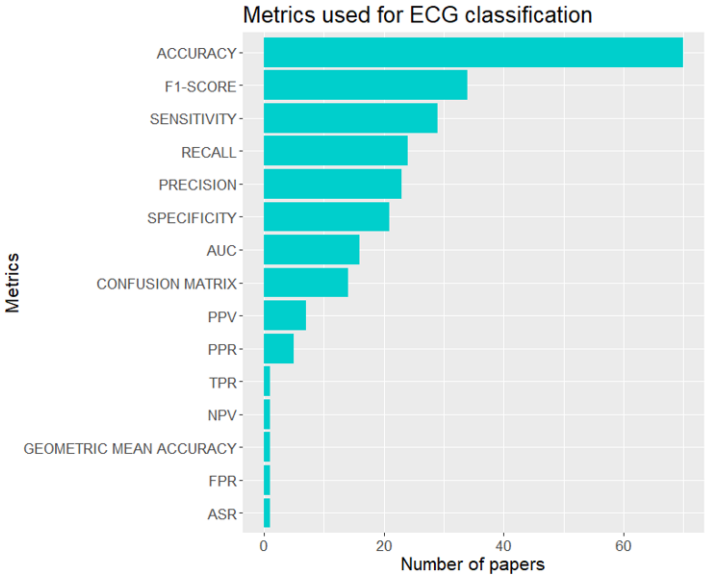


Figure 3.4. Most used metrics for ECG

From the previous chart it is evident that the most used measure is the **Accuracy**. In the second place, the *F1-Score* is often used. It is important to clarify that the latter is preferred when dealing with unbalanced data, since it take into account both *Recall* and *Precision* for its calculation. Some papers that consider the 4 mentioned measures at the same time can be found in [45], [55] and [25].

3.3 Federated learning for ECG

Once provided materials in [17] were assessed, I've compiled a list of key points about Federated Learning (FL) for ECG classification. I summarize the most important findings at each issue in the next paragraphs.

3.3.1 Methods for ECG classification using Federated Learning

In the ambit of Federated Learning (FL), most of the authors used Deep Neural Networks (DNN) and Convolutional Neural Networks (CNN). It has been noticed that traditional Machine Learning algorithms (like SVM, Random Forest, etc) are not usually employed for classify ECG signals in a FL context.

As an example, the authors of [45] used explainable artificial intelligence (XAI) and deep CNN to create a revolutionary end-to-end framework for ECG-based healthcare in a federated setting. With five-fold cross-validation, the trained classifier exceeded previous studies, attaining accuracy of up to 94.5 percent and 98.9 percent for arrhythmia diagnosis using noisy and clean data, respectively. In a federated

scenario, they also presented a new communication cost reduction strategy that reduces communication costs while improving the privacy of users' data. The reported results can be found on figure 3.5.

Scheme	Centralized or Federated	Acc (clean data)	Acc (noisy data)
[21]	Centralized	86.0%	–
[22]	Centralized	96.9%	–
[25]	Centralized	98.1%	–
[26]	Centralized	96.5%	–
[28]	Centralized	93.1%	–
[27]	Centralized	98.7%	–
[29]	Centralized	94.9%	–
[30]	Centralized	98.1%	–
[77]	Centralized	98.6%	–
[76]	Centralized	98.3%	–
[19]	Centralized	98.8%	–
Proposed	Federated	98.9%	94.5%

Figure 3.5. Comparison with previous studies for ECG classification [45]

To save critical communication bandwidth, the paper [47] adapted their suggested federated learning architecture for ECG analysis by asynchronously updating the shallow and deep model parameters of a proprietary CNN-based lightweight AI model. The results showed that the suggested asynchronous federated learning (Async-FL) approach can improve classification performance while simultaneously ensuring privacy, flexibility to new subjects, and reducing network bandwidth usage. Their proposed focus is in figure 3.6.

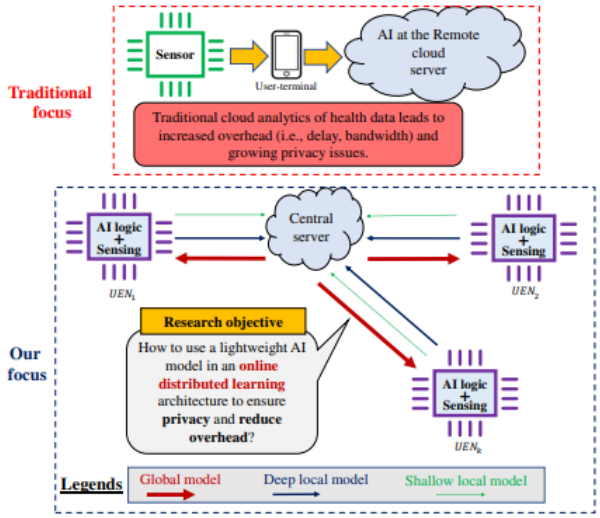


Figure 3.6. The main focus was to do asynchronously FL-based ECG method at the ultra-edge nodes (UENs) to classify ECGs preserving patient-data privacy[47]

In order for Federated Learning to be performed on low-capacity devices in real-world settings, the authors of [22] did an interesting job. For them, the training process must focus not only on achieving the highest level of accuracy, but also on lowering training time and resource consumption. In that study, they described a

model training method that incorporates a dynamic epoch parameter. In Federated Learning, they offered the BePOCH (Best Epoch) algorithm to determine the optimal number of epochs every training round. In studies with medical datasets, they showed that using the BePOCH recommended number of epochs reduces training time and resource consumption while maintaining accuracy.

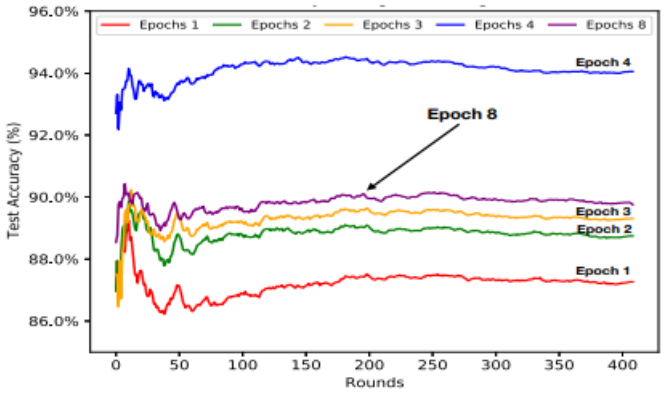


Figure 3.7. FL at various times throughout rounds. More epochs not always imply greater precision. Model with 8 epochs has a poorer accuracy than model with 4 epochs. [22]

3.3.2 Methods to handle NON-IID data

Along the literature, the researches performed have been focused on tackling the Non Independent Nor Identical distributed (Non-IID) problem that make the models in FL under-perform. In the following part I present a summary of the most relevant techniques used to deal with the mentioned issue.

Over/under-sampling: This is one of the most common techniques used to balance the data. It consist of randomly creating (or removing) data to equate distributions. The simplest way to do it is called Random Oversampling (ROS). The latter is the process of randomly picking and replacing instances from the minority class in the training dataset. There are other techniques like SMOTE [45]. SMOTE is a data augmentation algorithm that creates synthetic data points depending on the original data points. The technique can be thought of as a more advanced variant of oversampling or as a specific data augmentation process. SMOTE has the advantage of not creating duplicate data points, but rather synthetic data points that are somewhat different from the original data points [32].

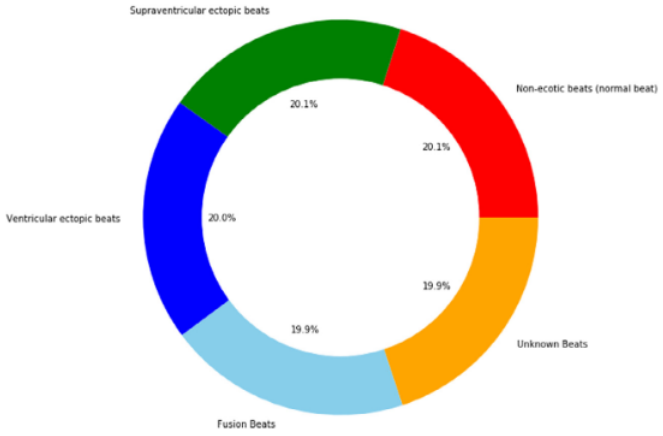


Figure 3.8. The distribution of the up-sampled (re-balanced) dataset. [45]

Data sharing strategy: In the paper [65] they showed that for neural networks trained with highly skewed non-IID data, where each client device trains just on a single class of data, the accuracy of federated learning drops by up to 55%. They also showed that the weight divergence, which can be measured by the Earth mover's distance (EMD) between the distribution over classes on each device and the population distribution, can explain the loss in accuracy. As a solution, the proposal was to establish a limited sample of data that is globally shared across all edge devices to improve training on non-IID data. Their proposed data sharing is exposed in the following image:

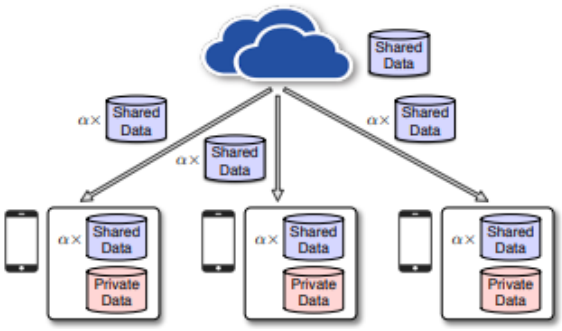
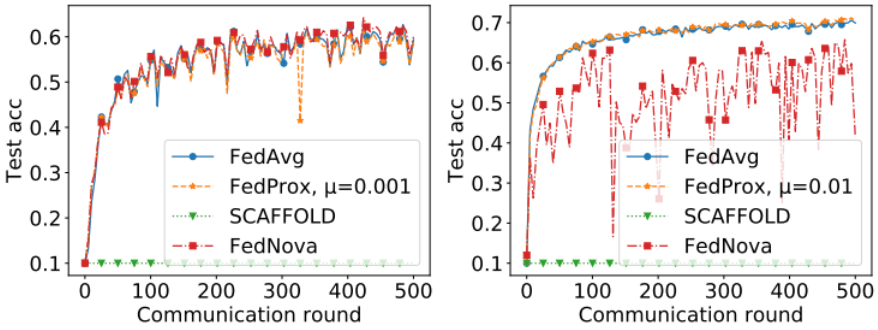


Figure 3.9. Illustration of the data-sharing strategy. [65]

Fedprox: FedProx uses Federated Average Aggregation (FedAvg) to improve the local aim. It restricts the size of local updates directly. To limit the distance between the local model and the global model, it adds an additional L2 regularization term to the local objective function. This is a simple approach to keep the local updates under control so that the averaged model stays close to the global optima. To regulate the weight of the L2 regularization, a hyper-parameter is introduced [30].

FedNova: FedAvg is improved during the aggregation stage. Varying parties may undertake different numbers of local steps (i.e., the number of mini-batches in the local training) each round, according to the model. When parties have varying processing power under the same time restriction, or when parties have various local dataset sizes under the same number of local epochs and batch size, this can happen [30].

SCAFFOLD: uses the variance reduction technique to model non-IID as introducing variance among the parties. It introduces control variate for the server and parties, which are used to predict the server model's update direction and each client's update direction. The difference between these two update directions is then used to approximate the drift of local training. By include the drift in the local training, SCAFFOLD corrects the local updates [30].



(a) $p_k \sim Dir(0.5)$ (b) $q \sim Dir(0.5)$

Figure 3.10. Training curves of different approaches on CIFAR-10 with 100 parties and sample fraction 0.1. [30]

Classifier Calibration with Virtual Representations (CCVR):It uses virtual representations sampled from an estimated Gaussian mixture model to modify the classifier. On popular federated learning benchmarks including as CIFAR-10, CIFAR-100, and CINIC-10, experimental findings show that CCVR achieves state-of-the-art performance [31].

	Method	$\alpha = 0.5$	$\alpha = 0.1$	$\alpha = 0.05$	CIFAR-100	CINIC-10
No Calibration	FedAvg	68.62±0.77	58.55±0.98	52.33±0.43	66.25±0.54	60.20±2.04
	FedProx	69.07±1.07	58.93±0.64	53.00±0.32	66.31±0.39	60.52±2.07
	FedAvgM	69.00±1.68	59.22±1.14	51.98±0.91	66.43±0.23	60.46±0.73
	MOON	70.48±0.36	57.36±0.85	49.91±0.38	67.02±0.31	65.67±2.10
CCVR (Ours.)	FedAvg	71.03±0.40 (↑ 2.41)	**62.68±0.54** (↑ 4.13)	54.95±0.61 (↑ 2.62)	66.60±0.63 (↑ 0.35)	69.99±0.54 (↑ 9.79)
	FedProx	70.99±1.21 (↑ 1.92)	62.60±0.43 (↑ 3.67)	**55.79±1.07** (↑ 2.79)	66.61±0.48 (↑ 0.30)	70.05±0.66 (↑ 9.53)
	FedAvgM	**71.49±0.88** (↑ 2.49)	62.64±1.07 (↑ 3.42)	54.57±0.58 (↑ 2.59)	66.71±0.16 (↑ 0.28)	**70.87±0.61** (↑ 10.41)
	MOON	71.29±0.11 (↑ 0.81)	62.22±0.70 (↑ 4.86)	55.60±0.63 (↑ 5.69)	**67.17±0.37** (↑ 0.15)	69.42±0.65 (↑ 3.75)
Oracle	FedAvg	72.51±0.53 (↑ 3.89)	64.70±0.94 (↑ 6.15)	57.53±1.00 (↑ 5.20)	66.84±0.50 (↑ 0.59)	**73.47±0.30** (↑ 13.27)
	FedProx	72.26±1.22 (↑ 3.19)	64.63±0.93 (↑ 5.70)	57.33±0.72 (↑ 4.33)	66.68±0.43 (↑ 0.37)	73.10±0.57 (↑ 12.58)
	FedAvgM	**73.30±0.19** (↑ 4.30)	64.24±1.32 (↑ 5.02)	57.11±1.08 (↑ 5.13)	66.94±0.32 (↑ 0.51)	72.88±0.37 (↑ 12.42)
	MOON	72.05±0.16 (↑ 1.57)	**64.94±0.58** (↑ 7.58)	**58.14±0.47** (↑ 8.23)	**67.56±0.44** (↑ 0.54)	73.38±0.23 (↑ 7.71)

Figure 3.11. Accuracy (%) on CIFAR-10 with different degrees of heterogeneity ($\alpha \in \{0.5, 0.1, 0.05\}$), CIFAR-100 and CINIC-10. [31]

Federated Cloning-and-Deletion (FedCD): It is a learning system that involves iterative cloning of global models at predetermined milestones, adaptive updating of a high-scoring subset of global models, and deletion of poor-performing models to produce a specialized model for each archetype. Devices can self-select into groups with similar data by maintaining various global models and updating models that perform well on their local data. This allows for faster convergence as well as increased accuracy [28].

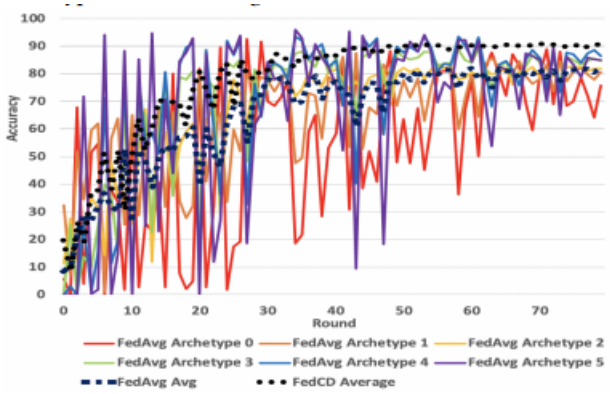


Figure 3.12. Comparisons of test accuracy for the FedAvg and FedCD (dotted) algorithms. [28]

Inverse Distance Aggregation (IDA): It is a new robust aggregation method that reduces inconsistency among updated local parameters caused by the NON-IID problem. The computation of the coefficients α_k, which is based on the inverse distance of each client parameter to the average model of all clients, lies at the heart of that method. This enables the poisoned models, i.e. out-of-distribution models, to be rejected or weighed less heavily [62].

Method	n_{cc}	Global Accuracy	Local Accuracy
FedAvg	1	**69.72**	60.52 ± 9.20
IDA	1	69.16	$\mathbf{61.21 \pm 8.79}$
FedAvg	2	**62.23**	57.14 ± 10.84
IDA	2	61.21	$\mathbf{60.21 \pm 5.48}$
FedAvg	10 (iid)	63.5	52.88 ± 15.73
IDA	10 (iid)	**63.72**	$\mathbf{57.38 \pm 10.56}$

Figure 3.13. Investigation on unbalanced data distro. among the clients in FL, with 5 random classes per client, and random number of samples per client for HAM10k. [62]

3.3.3 Metrics for Federated Learning

Inside the FL framework the same metrics showed in 3.2.3 are often used. In the ECG case, metrics like Accuracy, F1-Score, Recall and Precision are usually employed to measure the overall capacity of the model to detect the diagnoses. In addition, with the introduction of FL, some new metrics are considered to determine the behaviour of the training stage while considering the local nodes. Those metrics are explained in the following sections.

The authors of [11] provided a set of metrics for assessing individualized FL models in terms of performance and fairness. They computed the following metrics on the Quantum of Improvement (QoI) to quantify the per-user accuracy gains acquired in terms of personalization:

$$F_i = P_i - max(G_i, L_i) \tag{3.1}$$

where P, G, and L relate to the personalized model's accuracy, FedAvg and local model of user I while F_i refers to user i's QoI. In all equations, F will be referred as the QoI from now on.

The QoI can have unfavorable results. This suggests that the tailored method reduces the accuracy of a user's personalized model rather than increasing it as expected when compared to local or global models. In such instances, using evaluation measures directly may lead to erroneous results interpretation. As a result, there was a division of the QoI into two sets, each including the absolute QoI values: a set of positive QoI users (U+) and a set of negative QoI users (U-). The introduced measurements are then applied to both sets and interpreted accordingly.

Performance Metrics:

Percentage of User-models Improved (PUI): It is the percentage of users that see an improvement in their local and global models. A personalized model

should, in theory, increase the per-user accuracy of a large number of users.

$$PUI = \frac{COUNT(F_i)}{COUNT(U)} \times 100,\ i \in U\ (U : Users) \tag{3.2}$$

Median Percentage of Improvement (MPI): Calculated as Median(U+), where the *Median()* function returns the input's median and U+ is the QoI of the group of users who improved their performance. A tailored model should have a high median of QoI values among users who improve.

Average Percentage of Improvement (API): Measures the average percentage improvement among users who enhanced their performance (U+).

$$API = \frac{\sum_{i \in U^+} F_i}{len(U^+)} \tag{3.3}$$

In some cases, a customizing strategy does not improve users' local and global accuracy. In such instances, it is critical to disclose the personalized model's per-user accuracy decline. There was a definition of two metrics to evaluate the decreasing accuracy because this drop cannot be obtained from the improvement measurements (MPI and API).

Median Percentage of Decrease (MPD): Calculated in the same way as MPI: Median(U-).

Average Percentage of Decrease (APD): It is the average percentage reduction among users whose performance has reduced (U-).

Fairness Metrics

:

The aforementioned measures were extended to evaluate personalization strategies that produce better results in terms of fairness. Based on the relation reflected by the fairness metric, the QoI distribution among K users is more fair (uniform) under technique t than t' for two approaches t and t'.

Average Variance (AV): The AV is a measurement of data spread. It is defined as follows:

$$AV = \frac{1}{K} \sum_{i=1}^{K} (F_i(t) - \hat{F}(t))^2 \tag{3.4}$$

A lower AV indicates that a tailored technique can provides more fairness (uniformity).

Entropy: A measure that considers the magnitude of the QoI values. It can be calculated as:

$$Entropy = -\sum_{i=1}^{K} \frac{F_i(t)}{\sum_{i=1}^{K} F_i(t)} \log \left(\frac{F_i(t)}{\sum_{i=1}^{K} F_i(t)} \right) \tag{3.5}$$

A personalized technique with a bigger Entropy has a higher fairness potential.

Physical Metrics:

Some publications included metrics that deal with the physical part (with the actual devices). For example, [23] provided the following measurements:

CPU and Memory Consumption: Figure 3.14 shows the CPU and memory usage of a single worker node after 15 minutes of continuous training operations (epoch=10). Figure 3.14 shows that practically all four cores of the CPU are utilised when the client trains the local model for multiple epochs before transferring it to the server in one communication round (98 percent of CPU). The findings show that complex models with millions of parameters might be impossible to train on such devices.

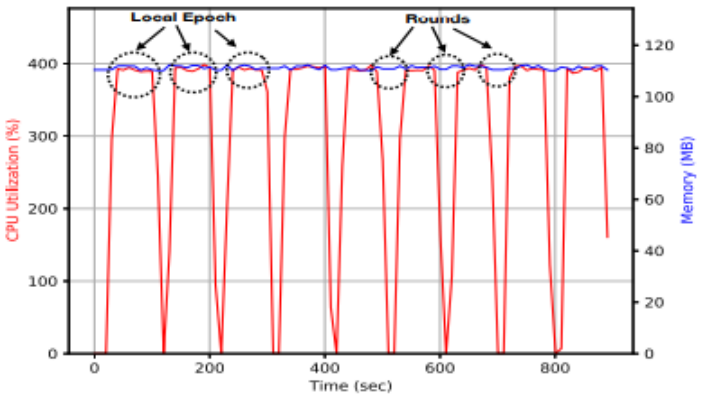


Figure 3.14. CPU and Memory Consumption

Training time and Temperature: When comparing the average training time for different numbers of workers, the training time increases significantly as the number of employees increments because the server waits for all clients to report back their freshly trained model (depicted in Figure 3.15).

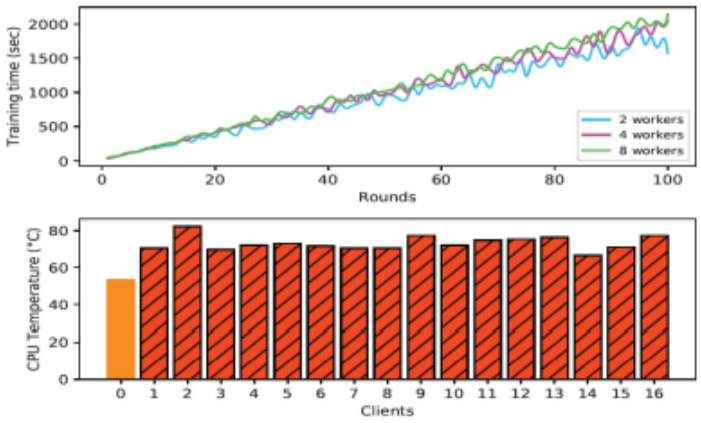


Figure 3.15. Training Time and Device Temperature

Chapter 4

Analytical techniques and tools

The parts that follow go over the most important analysis and approaches for studying, modeling, and predicting ECG arrhythmia diagnosis.

4.1 Commonly used techniques and tools

4.1.1 Data Wrangling (DW)

Data wrangling is the act of cleaning and combining chaotic and difficult data sets for easy access and analysis. With the amount of data and data sources growing all the time, it's more vital than ever to arrange massive volumes of data for analysis [44]. To facilitate data consumption and organization, this method normally requires manually transforming and mapping data from one raw format to another.

The most relevant Data Wrangling's objectives are [44]:

- Collect data from a variety of sources in order to uncover "deeper intelligence."
- As soon as feasible, get reliable, actionable data into the hands of business analysts.
- Reduce the amount of time it takes to collect and organize jumbled data before it can be used.
- Allow data scientists and analysts to focus on data analysis instead of data manipulation.
- Encourage senior executives in a company to improve their decision-making skills.

Figure 4.1. Main steps in Data Wrangling

The data wrangling approach typically consists of six iterative steps, as seen in Figure 4.1, as mentioned by [52]:

1. **Publishing:** Data wranglers prepare data for downstream usage - whether by a specific user or program - and identify any special actions or logic that were employed to do so.

2. **Discovering:** Before delving into the data, it's important to first have a better knowledge of what's there, since this will influence how you examine the data.

3. **Validating:** These are recurrent programming sequences that verify data quality, consistency, and security. Validation can include things like ensuring that qualities that should be distributed on a regular basis are distributed uniformly.

4. **Enrichment:** "What more types of data can be obtained from what already exists?" one can question during the data wrangling stage. or "What further information could assist me in making better selections based on the current data?"

5. **Structuring:** The data must be structured in this step of data wrangling because raw data arrives in a range of formats and sizes.

6. **Cleaning:** By altering null values and establishing standard formats, data wrangling aims to improve data quality.

4.1.2 Feature Engineering (FE)

The act of choosing, altering, and transforming raw data into features that may be utilized in supervised learning is known as feature engineering. It may be

necessary to build and train better features in order for machine learning to perform well on new datasets.

Challenge features

Within the Physionet 2020, the organizers provided a code that calculated 14 features leveraged on the recordings. Those variables where based on the R-Peaks and the RR interval.

R-Peaks: It refers to the R wave's highest amplitude (as seen in Figure 2.4).

RR-Interval: On an ECG, it is the period between two consecutive R-waves of the QRS signal. The former is determined by the sinus node's inherent features as well as autonomic factors.

Then, with the previous measures, the competence calculated the mean, median, standard deviation, variance, skewness and kurtosis ONLY for the first lead. In addition, the used the age and sex provided with the initial raw data.

Spectral features

Leveraged on the solution developed by [63], I implemented 636 features that deals with the spectral part of the signals provided in the ECG. Spectral analysis (where the spectral features were derived) is a frequently utilized tool for exploring biomedical data. The waveform component forms, their time positions within the cardiac cycle, and the regularity of the heart period all influence the ECG signal's spectrum ([53]).

Usually the Fourier Transform (FT) is used to extract information from signals like ECG. Nevertheless, the Fourier Transform has the drawback of capturing global frequency information, or frequencies that are present throughout a whole signal. This type of signal decomposition may not be appropriate for many applications, such as electrocardiography (ECG), which involves signals with short periods of distinctive oscillation. The Wavelet Transform, which decomposes a function into a set of wavelets, is another option that corrects the FT approach [54].

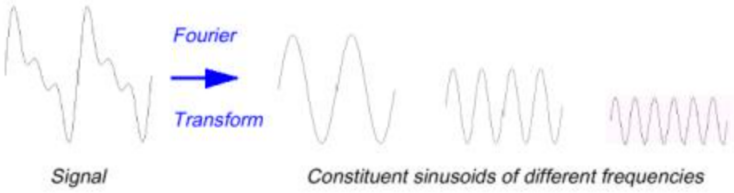


Figure 4.2. Wavelet representation

A Wavelet is a time-localized wave-like oscillation; an example is shown in Figure 4.2. Scale and location are the two most basic features of wavelets. The scale (or dilation) of a wavelet determines how "stretched" or "squished" it is. This attribute has to do with how waves are characterized in terms of frequency. The wavelet's position in time is defined by its location (or space).

Then, the schema of features calculated is as follows. For each lead calculate:

1. **Statistics**: Percentiles (5, 25, 50, 75, 95), mean, standard deviation and variance for the complete signals.

2. Calculate **coefficients** of **Discrete Wavelet Transform (DWT)**. DWT gets local frequencies for the signals. The Coefficients are calculated using the function `wavedec` from the Python's library `pywt`.

3. For each **coefficient** of DWT calculate:

    - **Statistics**: Percentiles (5, 25, 50, 75, 95), mean, standard deviation and variance.

    - **Shannon's entropy (same that entropy)**: It's related to the "amount of information" of a variable. In other words, it measures information of the distribution.

4.1.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis refers to the critical process of performing initial investigations on data to identify patterns, spot anomalies, test hypotheses, and check assumptions using summary statistics and graphical representations. It's important to initially comprehend the data before attempting to get as many insights as possible. EDA is all about making sense of data before getting their hands dirty with it. The major steps commonly examined in an EDA are shown in Figure 4.3.

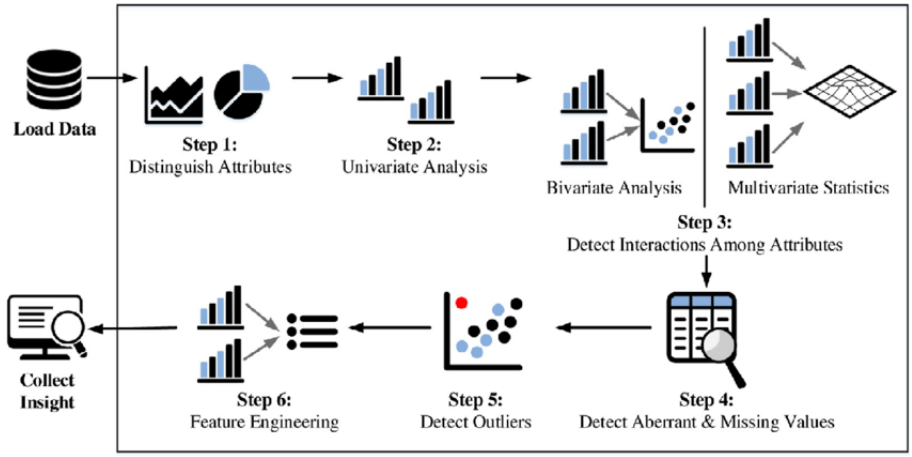


Figure 4.3. Schema of a EDA

4.1.4 Unbalanced classes

One of the most difficult issues when training a model is modeling imbalanced data [7]. When dealing with classification problems, the intended class balance is quite important. When a dataset has an uneven distribution of classes, the models attempt to learn only the dominant class, resulting in biased predictions.

One approach for addressing this issue is random sampling. Random resampling can be accomplished in two ways, each with its own set of benefits and drawbacks:

- **Oversampling:** Replicating examples from the minority class.
- **Undersampling:** Deleting examples from the majority class.

To put it another way, both oversampling and undersampling include creating bias by selecting more instances from one class than from another. The prior is used to compensate for an imbalance that is already present in the data or that is likely to occur if a perfectly random sample is obtained [13]. Because it makes no assumptions about the data, random sampling is a naive strategy. To minimize the data's influence on the Machine Learning algorithm, a fresh adjusted version of the data with a new class distribution is generated.

Random Oversampling and SMOTE were the two oversampling techniques chosen for this project. Synthetic Minority Oversampling Technique is a technique for creating synthetic samples for the minority class. Overcoming the problem of overfitting produced by random oversampling is easier with this method. It focuses on the feature space in order to generate new examples by interpolating between positive occurrences that are near in proximity.

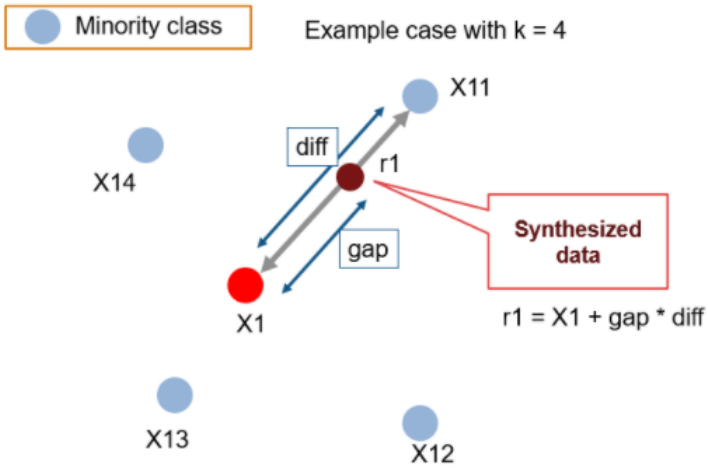


Figure 4.4. SMOTE process illustration

SMOTE uses the k-nearest neighbor technique to create synthetic data. To make them, it follows the instructions below. [7]:

1. Find the nearest neighbors of the feature vector.
2. Determine the distance between the two sample sites.
3. At random, the distance is multiplied by an integer between 0 and 1.
4. Find a new point on the line segment at the calculated distance.
5. Rep the procedure for each of the feature vectors that were discovered.

4.1.5 Machine Learning Models

Classifiers are the models provided in the following sections. These tools were created with the goal of determining which behaviors are more likely to be associated with various arrhythmia patterns. Each of these methods is widely utilized in various data-driven systems, and they have demonstrated useful behavior in a variety of classifying tasks, including ECG classification (3.2.2).

The various versions of the dataset were created using Python Notebooks in Google Colab. This section will detail the key models that were tested and evaluated.

Model 1 - (XGB) XG-Boost algorithm

The XG-Boost technique, which has proven to be effective in a variety of classification and regression problems, is the first attempt to classify the ECG signals. The aforementioned algorithm has been used to a variety of sectors, including economics,

credit rating, and health-related difficulties. The preceding are reasons to expect that such a strategy will be effective in the field of arrhythmia detection today.

XG-Boost is a decision-tree-based ensemble Machine Learning approach that uses gradient boosting ([8] [3]). When it comes to unstructured data prediction, *Artificial Neural Networks* outperform all other algorithms or frameworks (text, audio, pictures, etc.). However, for small-to-medium tabular data, such as the one utilized in this challenge, *decision tree-based* algorithms are now rated best-in-class.

XG-Boost minimizes a loss function to provide an additive expansion of the objective function, similar to gradient boosting. Because XG-Boost is only interested in decision trees as base classifiers, the complexity of the trees is controlled using a variation of the loss function.

$$L = \sum_{i=1}^{n} L(y_i, \hat{y_i}) + \sum_{k=1}^{K} \Theta(p_k) \tag{4.1}$$

$$\Theta(w) = \gamma Z + \frac{1}{2}\lambda ||w||^2 \tag{4.2}$$

The number of leaves on the tree is Z, and the leaf output scores are w ([3]). This loss function can be included into the split criterion of decision trees, resulting in a pre-pruning strategy. Trees with a greater γ value are easier to understand. The amount of loss reduction gain required to separate an internal node is determined by γ ([8]). Shrinkage is a regularization parameter in XG-Boost that decreases the step size in the additive expansion. Finally, other techniques such as tree depth can be utilized to keep the trees from becoming too complex. As a result of lowering tree complexity, the models are trained faster and need less storage space.

Model 2 - (Catboost) Catboost

The second candidate in predicting the arrhythmia type for ECG is the Catboost algorithm. The latter is a decision tree gradient boosting technique. It was created by Yandex (with its final version in 2017) researchers and engineers and is used by Yandex and other firms such as CERN, Cloudflare, and Careem taxi for search, recommendation systems, personal assistant, self-driving cars, weather prediction, and many other activities. Anyone can use it because it is open-source.

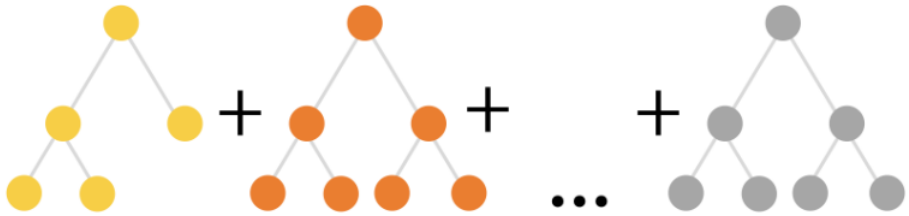


Figure 4.5. Catboost (decision trees) illustration

The implementation of ordered boosting [42], a permutation-driven alternative to the conventional approach, and a novel technique for processing category characteristics are two key algorithmic innovations offered in CatBoost. Both strategies were developed in order to combat a prediction shift induced by a specific type of target leakage found in all current gradient boosting algorithm implementations.

Model 3 - (DNN) Deep Neural Networks

A Deep Neural Network is another method for predicting ECG diagnosis. A DNN is a set of algorithms that attempts to recognize relationships in a batch of data by mimicking how the human brain functions.

In this context, deep neural networks refer to organic or artificial systems of neurons ([2]). Deep neural networks can adapt to changing input and produce the best possible result without requiring the output criteria to be modified because they can adapt to changing input. Neural networks, an artificial intelligence-based concept, are swiftly gaining popularity in the development of trading systems.

Neural networks aid in time-series forecasting, algorithmic trading, securities classification, credit risk modeling, and the generation of proprietary indicators and price derivatives in the financial world ([14] [41]). The deep neural network of the human brain is akin to a neural network. A "neuron" in a deep neural network is a mathematical function that collects and categorizes data according to a set of rules. The network closely resembles curve fitting and regression analysis, two statistical methods.

Perceptrons are grouped in interconnected layers in a multi-layered perceptron (MLP) [41], as indicated in Figure 4.6 . The input layer is responsible for collecting input patterns. In the output layer, input patterns can be mapped to classifications or output signals. Hidden layers fine-tune the input weightings until the neural network's margin of error is as little as possible. Hidden layers are supposed to deduce salient elements from input data that have the ability to predict outcomes. This is how feature extraction works, and it's similar to how statistical methods such as principal component analysis function ([41]).

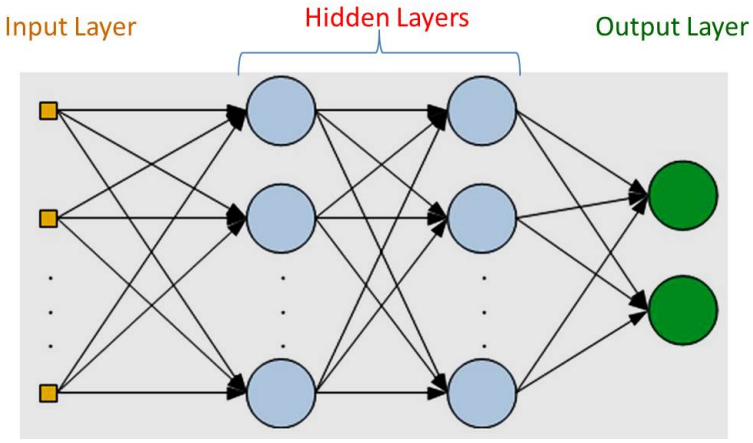


Figure 4.6. Deep Neural Network (Multi-layer Perceptron) schema

Model 4 - (LSTM) Long-Short Term Memory

Long short-term memory networks, are a type of Deep Learning network. It's a class of recurrent neural networks (RNNs) that can learn long-term dependencies, which is useful for solving sequence prediction issues. Apart from single data points like photos, LSTM has feedback connections, which means it can process the complete sequence of data.

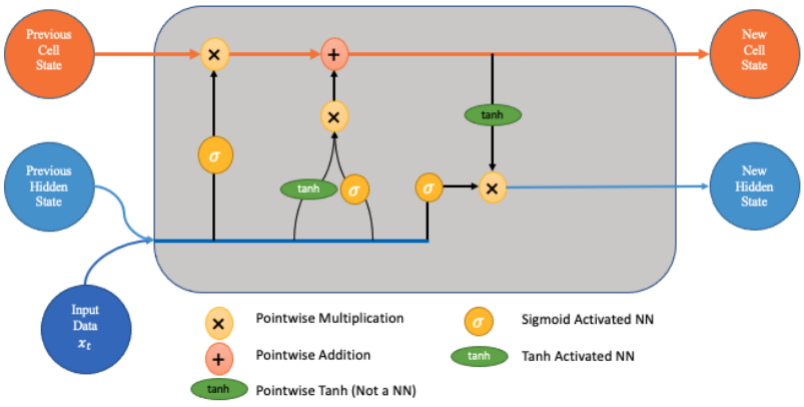


Figure 4.7. LSTM general schema

An LSTM model's primary role is played by a memory cell called a 'cell state,' which maintains its state across time. The horizontal line that runs through the top of the diagram below represents the cell state. It can be compared to a conveyor belt on which data just passes, unmodified [19].

4.1.6 Metrics

It is vital to create metrics that will assist in determining whether a model is better than others in order to determine whether it is better than others. There are explanations for each of the metrics used in the following sections.

Confusion Matrix

A confusion matrix, like the one shown in table 4.1, demonstrates how well a classification model works on test data for which the true values are known ([6]). The confusion matrix is simple in itself, but the related nomenclature can be confusing. In the following examples, I've created a hypothetical target variable called "Diagnose A" with the values "Yes" (if the recording belongs to that diagnose) and "No" (if the recording does not belong to that diagnose).

	Actual Class	
Predicted Class	Diagnose A - YES = 1	Diagnose A - NO = 0
Diagnose A - YES = 1	True Positives (**TP**)	False Positives (**FP**)
Diagnose A - NO = 0	False Negatives (**FN**)	True Negatives (**TN**)

Table 4.1. Confusion Matrix representation

Here is an explanation for each of the matrix's elements to understand the preceding terminology ([6] [20]).

- *True negatives (TN):* The model predicted they wouldn't have the diagnose A, and they don't.
- *True positives (TP):* These are examples when the model predicted yes (the recording has the diagnose A), and they actually don't.
- *False positives (FP):* The model projected that they would have the diagnose A, but they don't. (This is also referred to as a "Type I error.")
- *False negatives (FN):* The model anticipated that they would not have diagnose A, yet they do. (This is often referred to as a "Type II error.")

Accuracy

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{4.3}$$

The most basic performance metric is accuracy, which is defined as the proportion of correctly predicted observations to all observations. If a model is correct, one

would assume it is the best. Yes, accuracy is a relevant measure when the datasets are symmetric and the number of false positives and false negatives is about equal.

Imagining the case when the training set contains 98 percent samples of class A and 2% samples of class B, for example. The model may thus easily attain a 98% training accuracy by simply guessing every training sample that belongs to class A. When the same model is tested on a test set that contains 60% class A samples and 40% class B samples, the test accuracy reduces to 60%. As a result, classification accuracy is poor, but it gives the image of great accuracy.

Then, when the cost of misclassification of minor class samples becomes significant, ([20]) the true issue appears. The cost of failing to diagnose, for example, a sick person's ailment is significantly greater than the expense of submitting a healthy person to additional tests when dealing with a rare but lethal disorder.

Precision

$$Precision = \frac{TP}{TP + FP} \tag{4.4}$$

Precision [6] is the ratio of accurately predicted positive observations to total expected positive observations. This measure answers the question of how many of the drivers who were identified as drowsy actually drove. Precision is linked to a low false-positive rate.

Precision is a good statistic to employ when the costs of False Positive are high. Take, for example, the identification of email spam. In email spam detection, a false positive happens when an email that is not spam (actual negative) is wrongly identified as spam (predicted spam). If the precision of the spam detection model is low, the email user may miss important emails.

Recall

$$Recall = \frac{TP}{TP + FN} \tag{4.5}$$

Recall [6] is the ratio of successfully predicted positive observations to all observations in the actual class. It's meant to answer the question of how many drivers who actually slept were labeled as such.

In the case of identifying sick patients, for example, if a sick patient (Actual Positive) conducts the test and is predicted to be healthy (Predicted Negative). The cost of False Negative will be quite high if the condition is infectious.

F1 Score

$$F1Score = \frac{2 * (Recall * Precision)}{(Recall + Precision)} \tag{4.6}$$

The F1-Score is the weighted average of Precision and Recall. As a result, both false positives and false negatives are taken into account in this score. F1 is often more valuable than accuracy, despite the fact that it is less intuitive ([20] [6]). This is especially true if the class distribution is unequal. When the costs of false positives and false negatives are equal, accuracy works well. If the cost of false positives and false negatives differs significantly, it is best to evaluate both Precision and Recall.

4.2 Federated learning (FL) fundamentals

4.2.1 Definition

Federated learning (FL) is a machine learning technique for training machine learning models cooperatively on several devices or local servers in a decentralized way, preserving data privacy and data ownership for the device/server owner [12]. FL is extremely advantageous for highly decentralized healthcare data, especially with the growing prevalence of IoT devices for continuously capturing data and monitoring health.

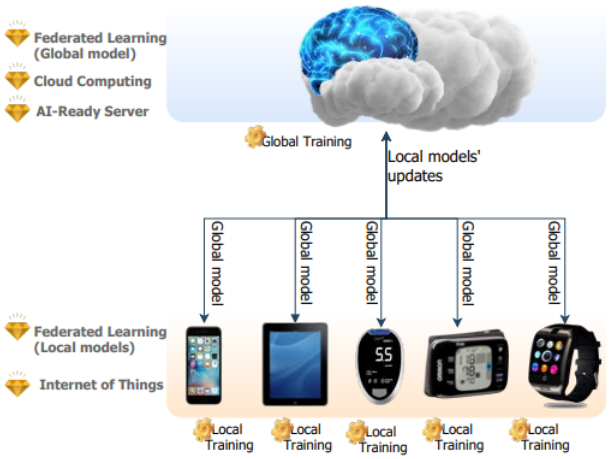


Figure 4.8. FL framework Overview

Figure 4.8 depicts a high-level view of the framework and how the technologies will interact together. The IoT devices will collect data from users and train a local deep learning model that is a copy of a global model that was previously received. Following the completion of the local training phase, the models will collaborate

to train a global model utilizing their updates rather than the raw data provided by the users. These model updates indicate changes in the weights of the models during the training process and do not reflect any private or personal information about the users.

All participating models will send updates to a cloud server, where they will be compiled and used to train the global model [47] [64]. Each device will receive a new copy of the updated global model once the global model training procedure is completed. As a result, the models will be trained and updated on a regular basis without sharing any personal information. As a result, the framework will support an IoT-based decentralized architecture in which models are spread among IoT devices without the need for a centralized server to operate the model and serve users. It will also protect users' privacy by processing and analyzing their data on IoT devices without disclosing it.

4.2.2 FL types

FL is divided into five categories [27] based on data partitioning, machine learning models (ML Models), privacy mechanisms, communication architecture, and federation scale.

Data Partitioning

The datasets of various clients share the same properties in **Horizontal data partitioning** [61], however there is limited sample space intersection. All FL architectures use horizontal partitioning the most. Aggregation at the server is made easier by the fact that a standard model can be used for all clients; FedAvg is typically used for aggregation. A dataset containing ONLY breast cancer patients from a specific hospital would be a simple to comprehend example.

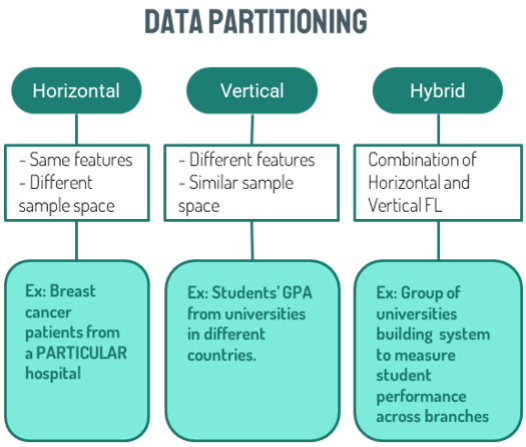


Figure 4.9. Data partition-based FL types

When clients are exposed to distinct feature spaces but the same or similar sample space, **Vertical Data Partitioning** comes into play. Entity alignment algorithms are utilized to find overlapping samples among the client data, and this overlapped data is used for training [27]. A dataset of students' GPAs obtained from institutions across the globe is a nice example. The feature space, which includes the grading scale and evaluation measure, is distinct.

Horizontal and vertical data partitioning are combined in **Hybrid Data Partitioning**. A set of universities intending to develop a FL System to assess student achievement across branches is an easy to comprehend case for hybrid partitioning.

ML models

The issue statement and dataset are frequently used to determine the machine learning models to use [27]. One of the most widely used models is neural networks (NN). Apart from NNs, decision trees are also used, as they are highly efficient and simple to understand. Models can be **homogeneous** or **heterogeneous** in a FL system.

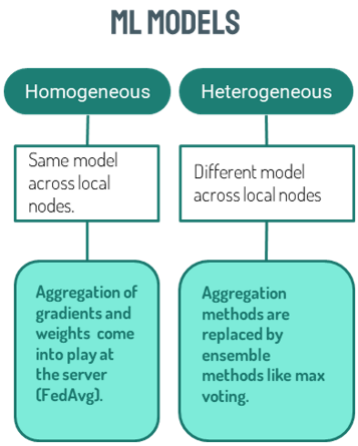


Figure 4.10. ML models-based FL types

In the case of the former, all clients use the same model, while the server uses gradient aggregation. In the latter instance, however, there is no possibility of aggregating because each client has a unique model. Aggregation methods are substituted with ensemble methods like max voting at the server in the case of heterogeneous models [27].

Privacy Mechanisms

The most controversial part of FL is how it deals with privacy. The main concept is to prevent client information from leaking out. The server may decipher the

data of clients without encryption by applying learning gradients. As a result, it's critical to hide the gradients. Differential privacy and cryptographic approaches are commonly used to address privacy concerns in FL systems.

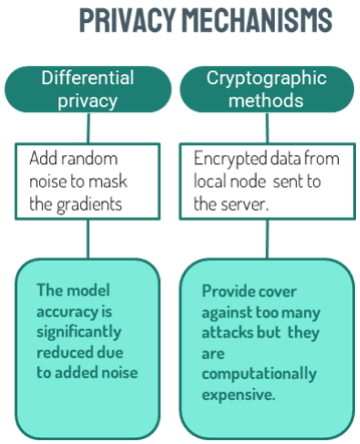


Figure 4.11. Privacy Mechanisms-based FL types

Differential privacy is a technique for hiding gradients by adding random noise to data or model parameters. Due to the extra noise, this strategy has a considerable negative in terms of model accuracy.

In FL systems, **cryptographic approaches** such as homomorphic encryption and safe multi-party computation are commonly used. The process is straightforward: clients send encrypted data to the server, the server processes the data, and then the encrypted output is decrypted to obtain the final result. Despite the fact that these methods provide protection against a wide range of threats, they are computationally intensive.

Architecture

There are two types of FL system architecture: centralized and decentralized. Both types of architecture work in the same way; the only difference is in client-server communication. We have a second model that acts as a server in a **centralized architecture**, and all parameter updates are done in this global model.

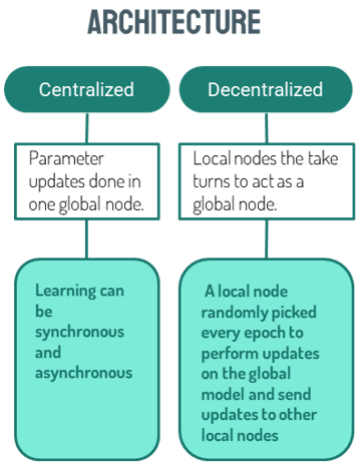


Figure 4.12. Architecture-based FL types

In a **decentralized design**, on the other hand, clients take turns acting as servers. Every epoch, a client is chosen at random to make global model changes and send the global model to other clients.

Scale of Federation

The scale of federation can be divided into two types: cross-silo and cross-device. To grasp the distinction between the two, relate cross-silo with organizations and cross-devices with mobiles. When using **cross-silo**, the number of clients is usually minimal, but they have a lot of computing power.

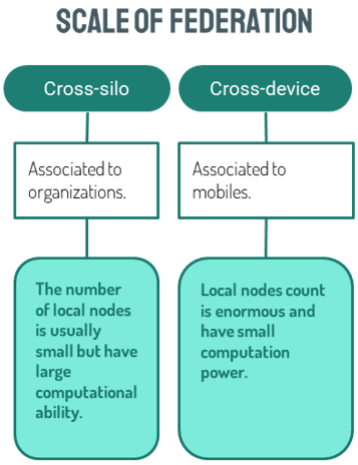


Figure 4.13. Scale of federation-based FL types

Regarding **cross-device**, the number of clients is enormous, but their computing power is limited. Another consideration is reliability: while we can rely on organiza-

tions (cross-silo) to be ready to train at all times, this is not the case with mobile phones (cross-devices). There's a chance that a bad network will make the gadget unavailable.

4.2.3 Advantages and disadvantages

FL has a lot of advantages over traditional, centralized systems [4]. Some of the most remarkable The upper hands of FL are:

- **Data security**: Keeping the training dataset on the devices eliminates the need for a data pool for the model.
- **Data diversity**: Companies may be unable to merge datasets from diverse sources due to challenges other than data security, such as network unavailability in edge devices. Federated learning makes it easier to access diverse data, even when data sources can only interact at particular periods.
- **Continuous learning in real time**: Models are continuously enhanced utilizing client input, eliminating the requirement to aggregate data for continuous learning.
- **Technology efficiency**: Because federated learning models do not require a single complex central server to evaluate data, this technique requires less complex hardware.

On the other hand, FL need to deal with some relevant challenges. The most common are:

- **Investment requirements**: FL models may necessitate frequent communication between nodes, which may necessitate an investment. This means that high bandwidth and storage capacity are among the system requirements.
- **Data Privacy**: In FL, data is not collected on a single entity/server; instead, numerous devices are used to collect and analyze data. Even though only models, not raw data, are transferred to the central server, models can be reverse engineered to identify client data, thereby increasing the attack surface. Differential privacy, secure multiparty computation, and homomorphic encryption are examples of privacy-enhancing technologies that can be utilized to improve the data privacy capabilities of federated learning.
- **Performance limitations**: In FL, models from several devices are combined to create a superior model. Device-specific factors may hinder the generalization of models from some devices, lowering the accuracy of the model's next

generation. Researchers investigated scenarios in which one of the federation's members could use secret backdoors in the joint global model to intentionally attack others.

4.2.4 Proposed approaches

In this study, I assume *a group of healthcare organizations* that wish to collectively train an arrhythmia classification AI-based module without sharing their medical records. For each organization, I envision a *single, local module* that coordinates all the activities related to the collection, storage and analysis of the medical records. Moreover, I assume that each organization has access to a set of high-definition electrocardiogram monitoring devices which are used to record the heart activity of the patients. Each monitoring session produces a short 12-channel ECG recording, i.e., about 10 seconds, that is transferred from the device to the local, private, database of the organization[1]. One or more healthcare experts examine the ECG recordings and provide a diagnosis that is also stored in the local database.

I assume that the group of healthcare organization have agreed upon a *single, global, trusted server*. The role of the global server is to coordinate all the activities of the local modules. The global server does not maintain any database with medical records. The only information stored is related to the common AI model and the operating parameters of the system. I also assume that all organizations have agreed on a common length for the ECG recordings, e.g., 10 seconds, and a common sampling frequency, e.g., $500KHz$[2,3].

Periodically, the global server starts *a global training session* by notifying all the local servers of the organizations that participate in the federation. Upon receiving this notification, each organization independently goes through *a local training session*. During a local training session, all the records available in the local, private, database are analyzed using local computing resources based on a *a processing pipeline* made up of four steps. First, each recordings is analyzed independently and key information connected to the heartbeats that will help the training of the AI model is extracted. Second, a *feature normalization* step follows where statistics are used to scale the features to improve the robustness of the data. In coordination with the global server, the local servers compute the necessary robust measures over the federated dataset without however revealing any sensitive information. Third, all

[1]Remark that several different interconnection architectures are used in ECG technologies available in the market and studied in the relevant literature, such as for example, wireless technologies like WI-FI or BLE, or wire technologies such as USB, or non-volatile memory formats. Such interconnections aspects are beyond the scope of this paper.

[2]Note that in case the recording have a different sampling frequency various algorithms exist in the relevant literature to change the sampling frequency to a lower or a higher one without affecting the accuracy.

[3]Note that ECG recordings that are longer than the agreed length can be split into multiple ones without loss of generality.

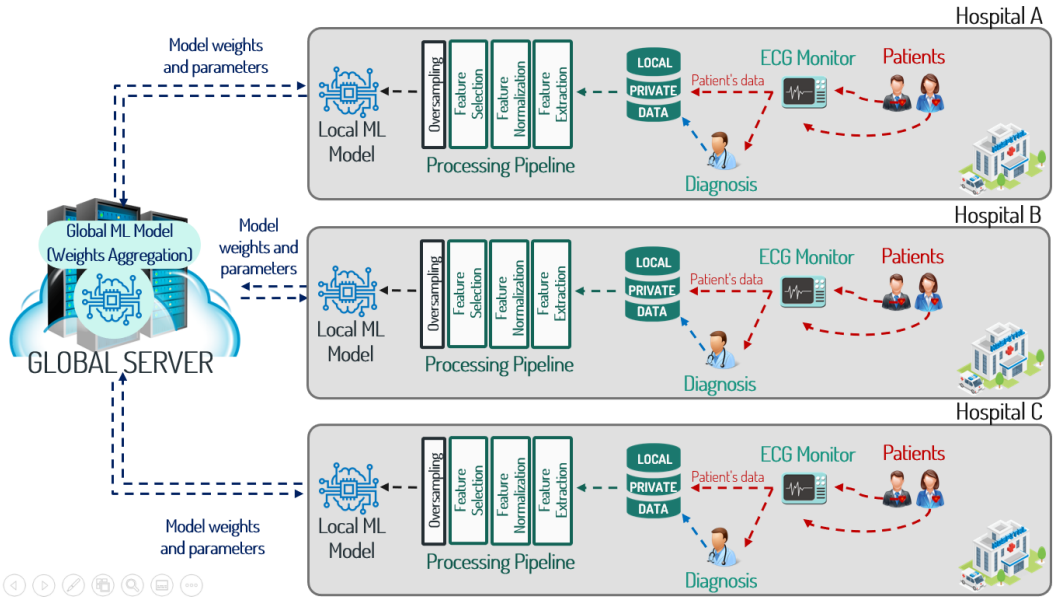


Figure 4.14. High-level overview of the proposed federated learning methodology and software and hardware components.

the original records along with the normalized features are examined and a *feature selection* is carried out with the goal to remove redundant features that may hinder the performance of the trained model and reduce the computational cost, due to the fact that they use a smaller number of features to train the local model. The fourth and final *data balancing* step examines the diagnosis attached to all the records in the local database to identify and remove any imbalances found between the representation of arrhythmia classes thus increase the generalization power of the model.

When all the local servers have completed the processing pipelines, under the coordination of the global server, they start training their local models. When the training is concluded, the weights of the resulting model are transmitted to the global server. The global server examines all the individual weights using a *weight aggregation method* and transmits the resulting model to all organizations. This process is repeated until either the *distributed optimization converges* or a certain number of steps is reached.

A high-level overview of the above methodology and the interconnection of the software and hardware elements that make up the federated architecture are depicted in figure 4.14.

After understanding the theory and characteristics behind the FL techniques I decided to test two different approaches using the PhysioNet 2020 datasets.

The first approach can be visualized in Figure 4.15. It is going to be called the Independent and Identically distributed **(IID) approach**.

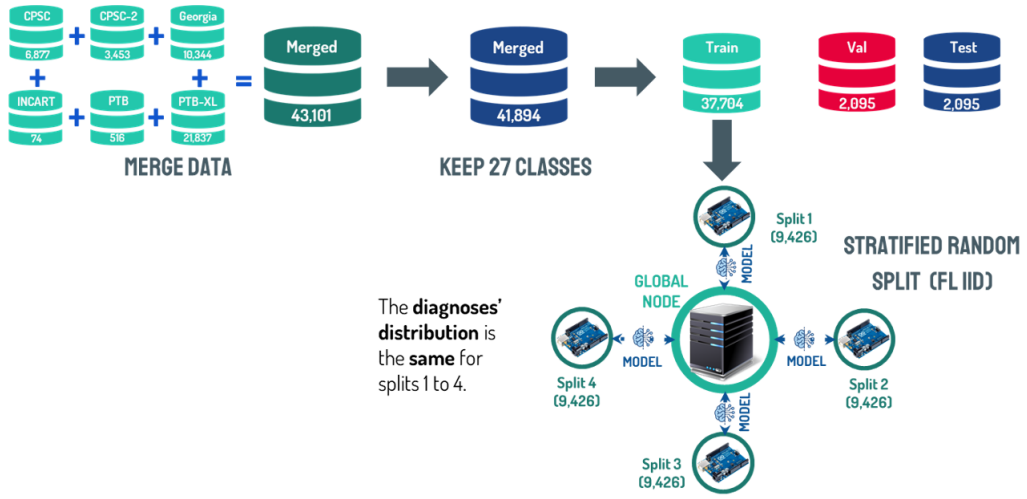


Figure 4.15. Approach 1: IDD approach

In the IID approach, I took the 6 databases (more information here 5.1) and appended them all in a single dataset with 43,101 recordings. After, using the analysis explained in figure 4.15, I ended up with 41,894 (filtering out the non-representative classes). Afterwords, I randomly split the data to get train, validation and test datasets. Afterwords, to get the IID splits I performed a Stratified random split, dividing the whole train data in 4 parts. In that way, the distribution of the labels (diagnoses) is the same for each partition.

The second (and more real) approach was called the **(Non-IID) approach**. It's structure is explained in figure 4.16.

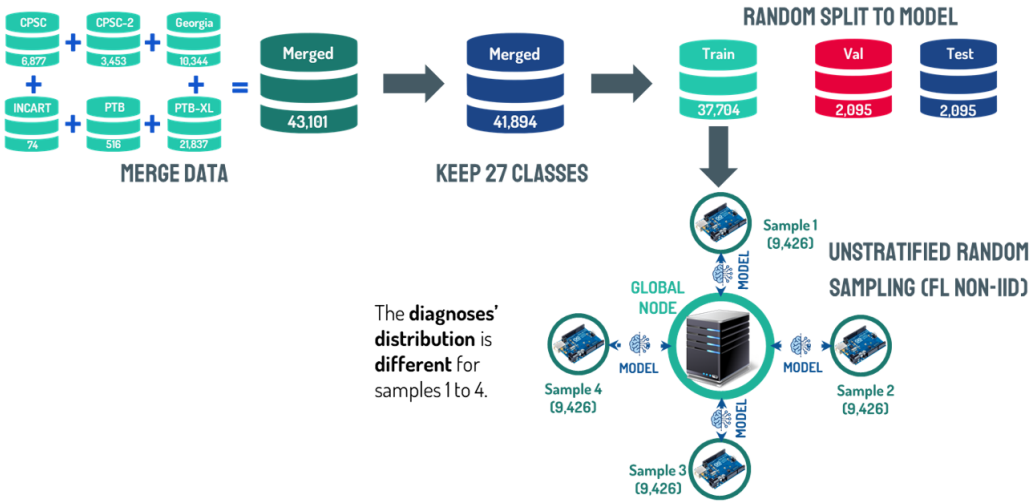


Figure 4.16. Approach 2: Non-IID approach

In the Non-IID approach, I combined all six datasets (additional information here

5.1) into a single dataset including 43,101 recordings. I came up with 41,894 after utilizing the methodology described in figure 4.16 (filtering out the non-representative classes). After that, I divided the data into train, validation, and test datasets at random. Later, I used an Unstratified random sampling (with replacement) method to acquire four separate samples from the original data. As a result, each sample has a varied distribution of labels (diagnoses).

Chapter 5

ECG Arrhythmias classification

5.1 Dataset employed

PhysioNet presents an annual series of biomedical 'Challenges' that focus on unsolved clinical and basic science challenges in collaboration with the annual Computing in Cardiology (CinC) conferences. The National Institutes of Health (NIH), Google, MathWorks, and the Gordon and Betty Moore Foundation have all lent their support to these challenges. George Moody, of the Laboratory for Computational Physiology (LCP), directed these Challenges for the first 15 years (from 2000 to 2014), before retiring due to ill health. Gari Clifford of Emory University and the Georgia Institute of Technology has been leading the Challenges since 2015. In 2021, the 'PhysioNet/Computing in Cardiology Challenges' were renamed the 'George B. Moody PhysioNet Challenges' to honor George's lifetime contributions to the discipline, particularly his seminal work on the Challenges [40].

The 2020 Challenge's purpose is to use 12-lead ECG records to detect clinical diagnosis. Starting from the clinical data provided, the participants must implement an open-source algorithm that can automatically classify the cardiac abnormality or abnormalities present in each 12-lead ECG recording and provide a probability or confidence score for each of them, with an emphasis on 27 diagnoses 5.1 To determine the winner, the trained models of the participants are run on hidden validation and test sets and their performance is evaluated using a novel, expert-based evaluation metric designed specifically for the 2020 Challenge. The team whose algorithm achieves the highest score is the winner of the Challenge.

Diagnosis	Code	Abbreviation
1st degree AV block	270492004	IAVB
Atrial fibrillation	164889003	AF
Atrial flutter	164890007	AFL
Bradycardia	426627000	Brady
Complete right bundle branch block	713427006	CRBBB
Incomplete right bundle branch block	713426002	IRBBB
Left anterior fascicular block	445118002	LAnFB
Left axis deviation	39732003	LAD
Left bundle branch block	164909002	LBBB
Low QRS voltages	251146004	LQRSV
Nonspecific intraventricular conduction disorder	698252002	NSIVCB
Pacing rhythm	10370003	PR
Premature atrial contraction	284470004	PAC
Premature ventricular contractions	427172004	PVC
Prolonged PR interval	164947007	LPR
Prolonged QT interval	111975006	LQT
Q wave abnormal	164917005	QAb
Right axis deviation	47665007	RAD
Right bundle branch block	59118001	RBBB
Sinus arrhythmia	427393009	SA
Sinus bradycardia	426177001	SB
Sinus rhythm	426783006	NSR
Sinus tachycardia	427084000	STach
Supraventricular premature beats	63593006	SVPB
T wave abnormal	164934002	Tab
T wave inversion	59931005	TInv
Ventricular premature beats	17338001	VPB

Table 5.1. Diagnoses, SNOMED CT codes and abbreviations for the 27 diagnoses that were scored for the Challenge.

The data are from five different sources:

1. CPSC Database and CPSC-Extra Database
2. INCART Database
3. PTB and PTB-XL Database
4. The Georgia 12-lead ECG Challenge (G12EC) Database
5. Undisclosed Database

The first source consists of three databases from the China Physiological Signal Challenge 2018 (CPSC2018), which took place in Nanjing, China at the 7th Interna-

tional Conference on Biomedical Engineering and Biotechnology [38]: the original public training dataset (CPSC), an unused dataset (CPSC-Extra), and the test dataset (the hidden CPSC set). The first two were shared as training sets, while the last one was split into validation and test set for the 2020 Challenge. This training set consists of two sets of 6,877 (male: 3,699; female: 3,178) and 3,453 (male: 1,843; female: 1,610) of 12-15 ECG recordings lasting from 6 seconds to 60 seconds. Each recording was sampled at 500 Hz.

The second source is the public dataset from the St. Petersburg Institute of Cardiological Technics (INCART) 12-lead Arrhythmia Database [15 V. Tihonenko, A. Khaustov, S. Ivanov, A. Rivin, and E. Yakushenko, "St Petersburg INCART 12-lead arrhythmia database", PhysioBank, PhysioToolkit, and PhysioNet, 2008, doi: 10.13026/C2V88N.]. The dataset was shared as a training set. This database consists of 74 annotated recordings extracted from 32 Holter records. Each record is 30 minutes long and contains 12 standard leads, each sampled at 257 Hz.

The third source from the Physikalisch Technische Bundesanstalt (PTB) includes two public databases which were shared as training sets: the PTB Diagnostic ECG Database and the PTB-XL, a large publicly available electrocardiography dataset. The first PTB database contains 516 records (male: 377, female: 139). Each recording was sampled at 1000 Hz. The PTB-XL contains 21,837 clinical 12-lead ECGs (male: 11,379 and female: 10,458) of 10 second length with a sampling frequency of 500 Hz.

The fourth source is the Georgia 12-lead ECG Challenge (G12EC) Database. This is a new database, representing a large population from the Southeastern United States, and is split between the training, validation, and test sets. The validation and test set comprised the hidden G12EC set. This training set contains 10,344 12-lead ECGs (male: 5,551, female: 4,793) of 10 second length with a sampling frequency of 500 Hz.

The fifth source is a dataset from an undisclosed American institution that is geographically distinct from the other dataset sources. This dataset has never been posted publicly and contains 10,000 ECGs all retained as test data [38]. As the mentioned dataset was not disclosed, I didn't use that one in my experiments.

The actual count of all the diagnoses by each database can be found in Figure 5.1. That was obtained by taking the first arrtyhmia reported by each recording since each patient could contain more than one diagnose based on its ECG.

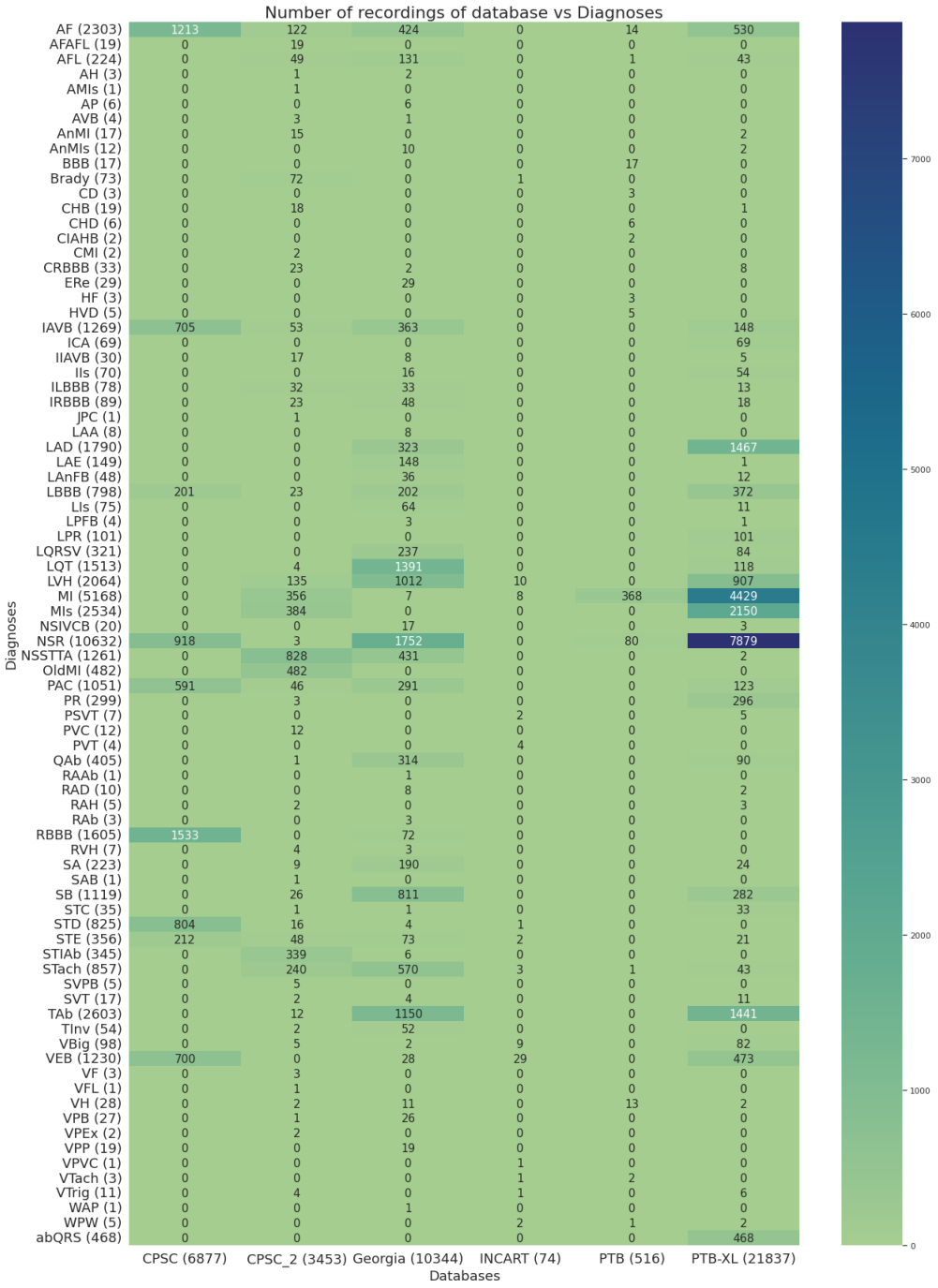


Figure 5.1. Number of recordings for each diagnosis by database

All data is provided in WFDB format. Each ECG recording has a binary **MATLAB v4 file** for the ECG signal data and a **text file** in WFDB header format describing the recording and patient attributes, including the diagnosis [39] [38] [15].

5.2 Centralized Learning

To get an understanding of the best performances achievable with the mentioned dataset I implemented a Centralized (or traditional) Learning. The latter means that I applied the analytical tools mentioned in 4 over the complete dataset, without dividing it into clients. Then the main processes and results are summarized in the following literals.

5.2.1 Data wrangling

As mentioned in 4.1.1, the data wrangling process is usually the first step when dealing with a data-oriented problem. In this case, I placed the data in a Google Drive folder after downloading it from the official competition's website [39]. Afterwords, using Google Colaboratory I extracted and organized the information in Python. The representation of the ECG along the 12-leads can be examined in Figure 5.2.

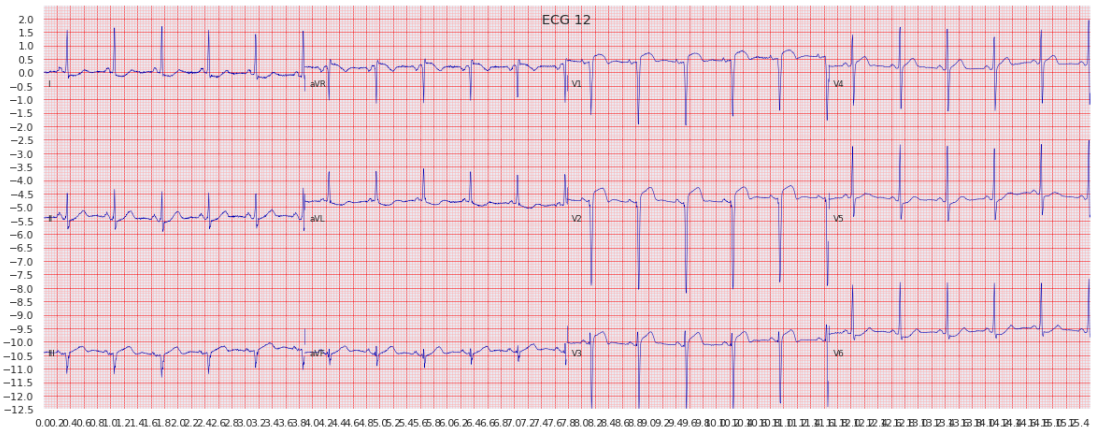


Figure 5.2. 12-lead ECG for recording S0033 of PTB database

Then, the whole data (all the databases) and the features mentioned in 4.1.2 where calculated. That process took almost 1 hour to run in an using the default configuration of Google Colab. Besides, for each recording I selected the first diagnose (arrhythmia) that appeared as the label to be predicted. The latter process ran in about 2 minutes.

5.2.2 EDA

Once the big dataset was loaded, it contained a total of 43,101 recordings and about 764 variables. From the latter, 650 where the features created and the remaining 3 corresponded to the id of the recording, the database that it belongs to and the label (response variable to be predicted). Then, the first analysis need was

to examine if the features contained any missing value. Using the function `bar` from the `missingno` library, I managed to explore the missing values. In figure 5.3 are depicted only the 50 first features' missing counts and percentage.

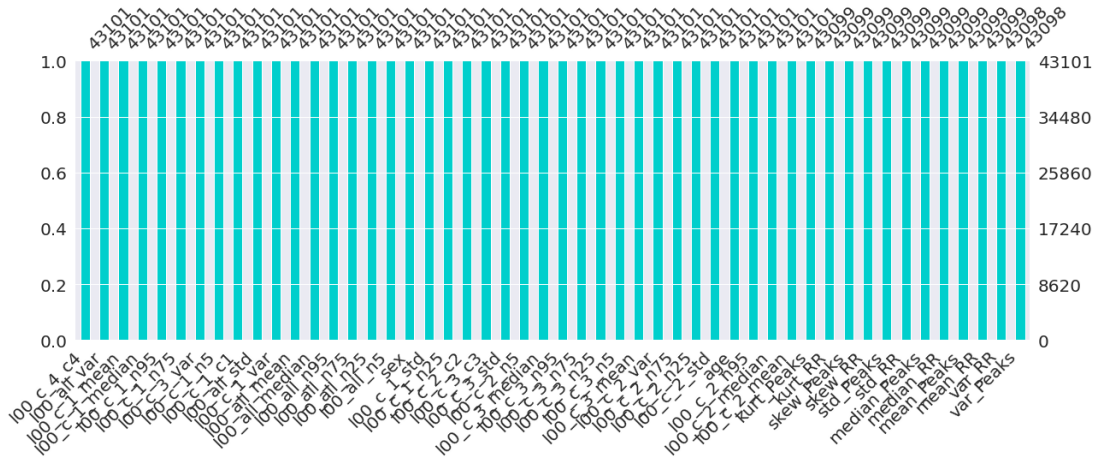


Figure 5.3. Missing counts and percentage for 50 features of the complete dataset

As shown in the previous chart, the missing percentage is considerably small (less than 0.1%). For that reason, I decided to impute those missing values by using the mean (average) of each attribute.

The second crucial aspect to investigate was the distribution of the response variable. Then, within the plot 5.4 it is shown the absolute count of each diagnose in the dataset.

Figure 5.4. Label distribution for the complete dataset

As evidenced in the previous chart, there are too many arrhythmias that don't have a big participation. That can lead to problems when trying to infer the predicted class of a recording, since there were not enough cases to learn the classifiers properly. That why those diagnoses with a participation smaller than 150 records where discarded from the analysis. With the previous filter, the selected data to work with got a size of 41,894 recordings distributed as shown in figure 5.5.

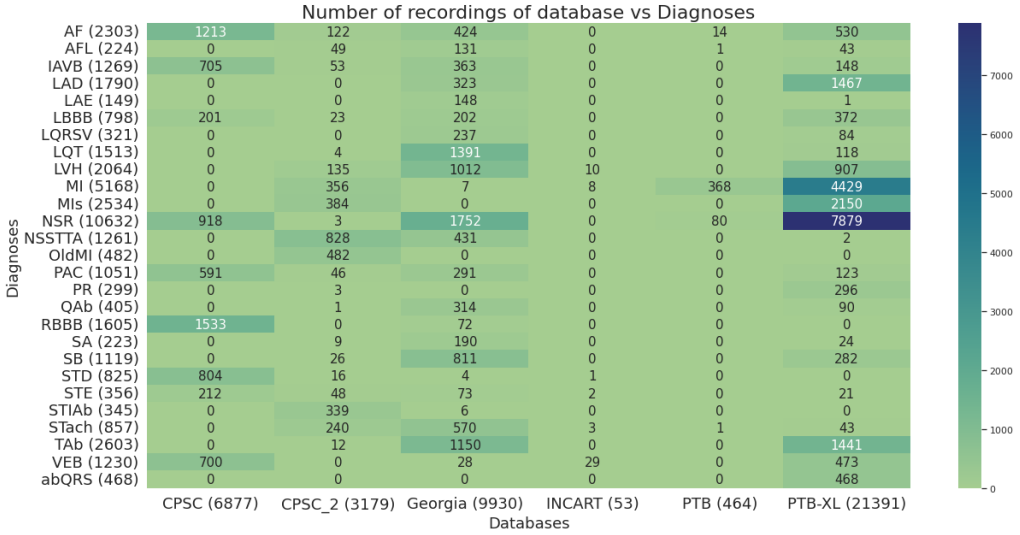


Figure 5.5. Number of recordings for each diagnosis by database for the filtered data

Besides, the final distribution of the labels ended up as shown in figure 5.6. As expected, the most common diagnose was Normal Sinus Rhythm (NSR), which is the normal status for a ECG. In addition. the arrhythmia with one of the smallest participation turned out to be Sinus Arrhythmia (SA).

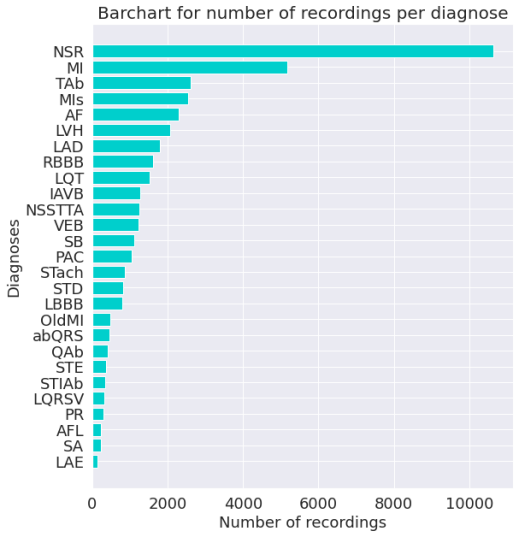


Figure 5.6. Label distribution for the filtered dataset

5.2.3 Feature Selection and normalization

In an analytical context, having a huge amount is a double-edged sword. On the one hand, the more information existing to predict a phenomena, the better. On the other hand, the computational time required to process to much information

may lead to training times that are not affordable. Regarding the latter I decided to perform a feature selection step in order to determine the most important features to predict the arrhythmias.

$$feature_s = l(lead_i)\_c\_(coefficient_j)\_(operation_k) \tag{5.1}$$

Each 636 spectral feature is based on *lead* as l, their *coefficient* as c and *operation* applied on it for example: mean, coefficient percentiles, standard deviation etc. Also, $lead_i$ represents ECG lead number from 00-11, $coefficient_j$ represents a coefficient number from Discrete Wavelet Transform(there are 5 coefficients in total 1-5) and $operation_k$ represents the operation name like mean as average, standard deviation as std, variance as var, percentiles represents as n5(percentile 5), n25(percentile 25), n50(percentile 50), n75(percentile 75), n95(percentile 95) and entropy applied on coefficient get represented as c1-c5. At the end from each ECG lead, we get 53 features and in 53x12 we get 636 spectral features in total. Below in figure [5.7] names of features are represented by above represented [5.1] morphology.

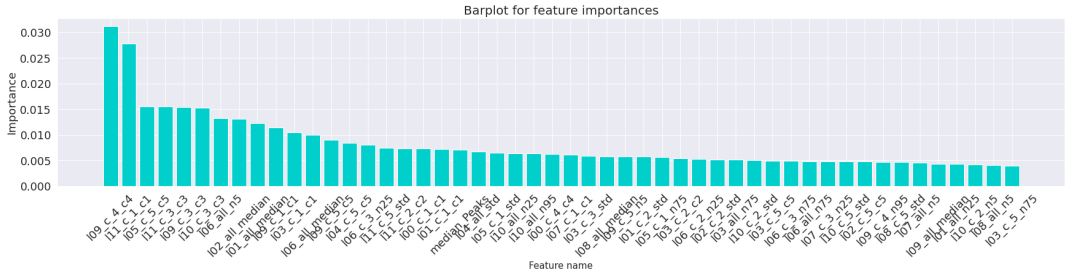


Figure 5.7. Feature importance from XG-Boost algorithm (only the 50 best)

The bar-plot in figure 5.7 depicted the most important features to predict the classes obtained by means of the XG-Boost method. The latter provides an automatic raking of the most relevant features to classifier the ECGs. I decided to take the best 120 variables since they managed to get a good enough accuracy, compared to the one obtained using all the features. The best features turned out to be the Entropy for leads: 9, 11, 10; the percentile 5% for lead 6; and the Median for leads 1, 2.

As an additional tool to enhance the performance of the models there was an implementation of features normalization. In this case I tried three different techniques to transform the features to the same scale. The approaches tried were provided by the `sklearn` library in Python. Those are: `StandardScaler`, `MaxMinScaler` and `RobustScaler`. In the end, the scenario that provided the best results was using `RobustScaler`. The latter uses statistics that are resistant to outliers to scale features. The median is removed, and the data is scaled according to the quantile range (defaults to IQR: Interquartile Range). The interquartile range

(IQR) is the distance between the first and third quartiles (25th and 3rd quantiles) (75th quantile).

5.2.4 Balancing classes (arrhythmias)

As depicted in 5.6, the diagnoses have a imbalanced characteristic. The latter means that each category has a different participation over the data. That could represent a problem in the performance of the classifiers that will be proposed.

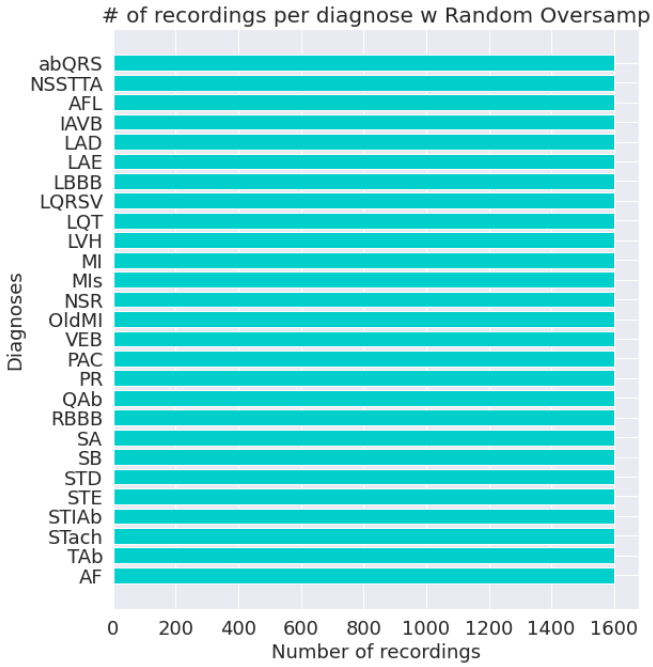


Figure 5.8. Number of recordings for each diagnosis by database for the ROS oversampled data

Then, two oversampling methods were proposed to deal with the imbalance issue. The first one is called **Random Oversampling (ROS)**. In the latter the minority classes are replicated together with its features. Besides, a down-sampling was applied to have a number of recording similar to the filtered dataset. In the end, the ROS dataset had 43,200 recordings. And as depicted in figure 5.8, the distribution of the labels is much more similar among the arrhythmia categories.

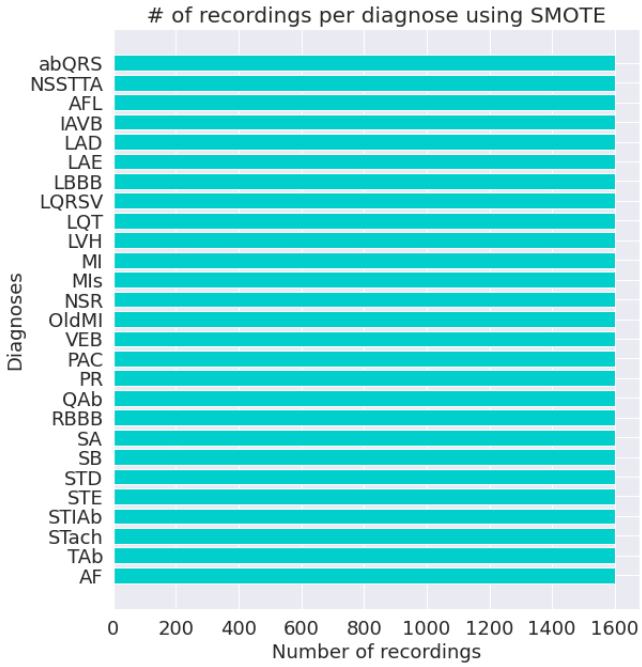


Figure 5.9. Number of recordings for each diagnosis by database for the SMOTE oversampled data

The second oversampling technique used was SMOTE (SMT) 4.1.4. In the end, the SMOTE dataset had also 43,200 recordings. And as shown in figure 5.9, the distribution of the labels is also similar among the arrhythmia categories.

5.2.5 Fitted models and results

With the previous pre-processing applied over the data, the following step was to adjust some Machine Learning models to the ECG's arrhythmias. During this step also other scenarios and considerations were employed [35]. A detailed explanation of the outlines is discussed in table 5.2.

Characteristic	Scenarios	Best approach
Data Split	%Train-%Validation-%Test: Option 1: 60%-20%-20% Option 2: 70%-10%-10% Option 3: 80%-10%-10% Option 4: 90%-5%-5%	Option 4: 90%-5%-5%
Features normalization	Option 1: MinMaxScaler Option 2: StandardScaler Option 3: RobustScaler	Option 3: RobustScaler
Sampling rate	Option 1: 257Hz Option 2: 500Hz	Option 1: 257Hz
Features employed	Option 1: Baseline features Option 2: Baseline features + Spectral features	Option 2: Baseline features + Spectral features

Table 5.2. Scenarios tried during modelling phase

Within the results in 5.10 it is highlighted that the best model is the one applied by team 2 of the Physionet competence, which obtained an F1-score close to 0.63. Nevertheless, the Deep Neural Network (DNN) over the ROS data had similar behaviour, having the mentioned metric in 0.61. Finally, LSTM does not perform that well compared to the other models since the metrics are between 0.39 and 0.47 for all the scenarios.

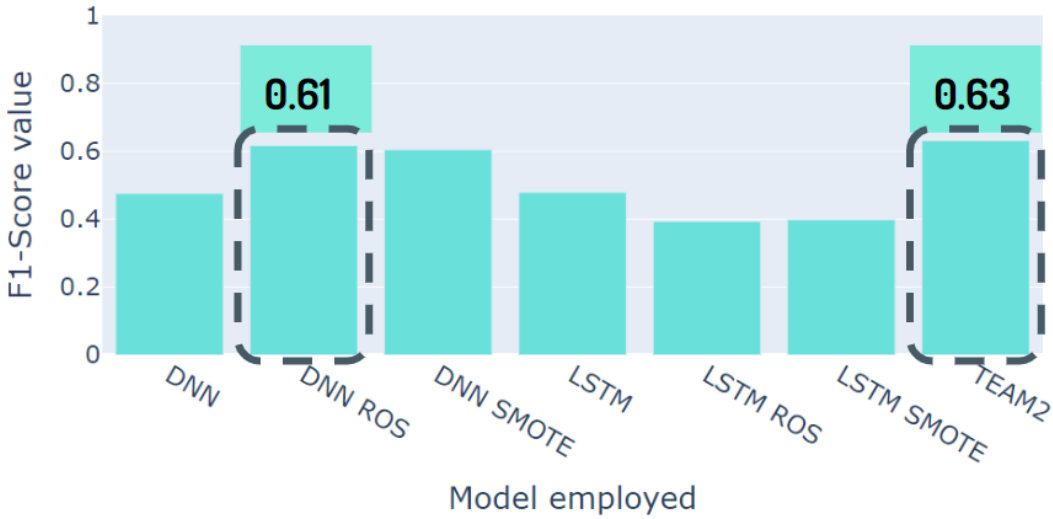


Figure 5.10. F1-Score for methods employed in Centralized Learning (CL) on the test set

A similar analysis derives from table 5.3, which shows the metrics employed in the study. Regards accuracy, TEAM2 got 0.64 and DNN attained 0.61, the latter applied over the ROS dataset. It is significant to clarify that using Accuracy is not

the best metric in this dataset since the labels are heavily unbalanced, which is why we use F1-score as a proper choice to compare the model's behaviour.

Figure 5.11. Execution times for the methods used in CL

Finally, we included a metric of the time to preprocess and train each model. As shown in figure 5.11, the TEAM2 approach took almost 122 minutes to run. On the other hand, DNN and LSTM took lower execution times (close to 89 minutes on average). Thus, TEAM2 is the slowest method, although it generates the best results. On the contrary, DNN is a fast method, and the performance is NOT quite different from TEAM2.

5.3 Federated Learning

Leveraged on the Centralized Learning method mentioned, it is time to immerse in the Federated Learning (FL) approach executed for this ECG dataset. With the CL it was possible to get an overall performance with the best techniques and scenarios to be applied. To deal with the FL proposed, there were two possible ways to work with the database in a Federated context. Those possibilities will be explained in the following chapters.

5.3.1 IID approach

This approach was based on the idea of using the whole dataset containing 41,894 registers and divide it in 4 different datasets (local nodes). As a parenthesis, the decision of the number of local nodes was based on selecting at least 30 diagnoses for the rarest arrhythmia. Due to the stratified random splitting, the mentioned data in each local node (or client) will be Independent and Identically Distributed

(IID) 4.2.4. This scenario is not completely realistic since usually the ECGs shared in multiple devices are Non-IID. Nevertheless, it is worth to try and see how the FL approach will perform over the data.

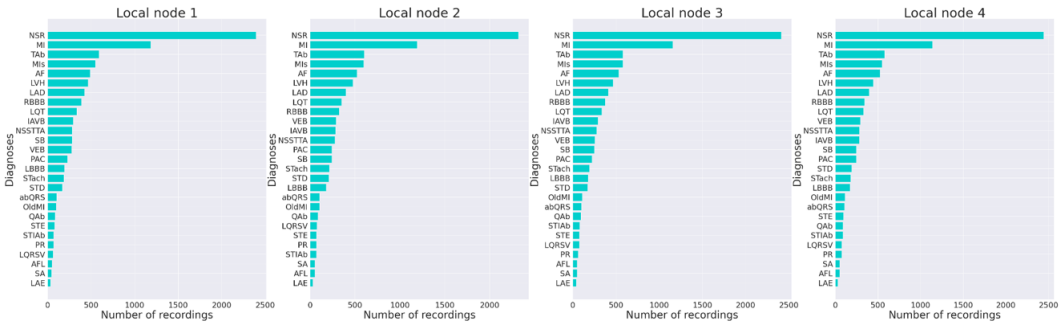


Figure 5.12. Label distribution for the filtered dataset by each local node

As depicted in figure 5.12, the distribution among the 4 local nodes seems IID. The later means that the diagnoses along the devices will be the same. Besides, each local node contains 9,426 recordings. The same occurs for the ROS and SMOTE datasets when dividing them in 4 clients, as is shown inf figures 5.13. Of course, in this case all the diagnoses have almost the same participation across the nodes, making them IID and balanced.

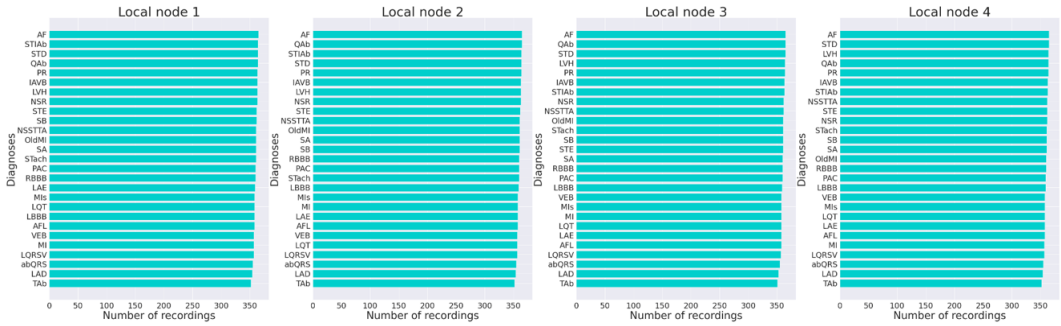


Figure 5.13. Label distribution for the ROS and SMOTE datasets by each local node

Once the four datasets were settled, the modelling part can be performed. As a reminder, in the FL technique, each local node will train a model and later it will send the weights to a global node where the weights are averaged and updated back in each client. Then, the DNN, LSTM and TEAM2 methodologies were used to classify the ECG's arrhythmias.

As depicted in figure 5.14, the best results arise using the TEAM2 method. The latter got an F1-score of 0.63 in the test set. On the other hand, DNN got an F1-score of 0.54, placing it as the second best option. In addition, the best performance for LSTM was obtained with the original data, although it is worst

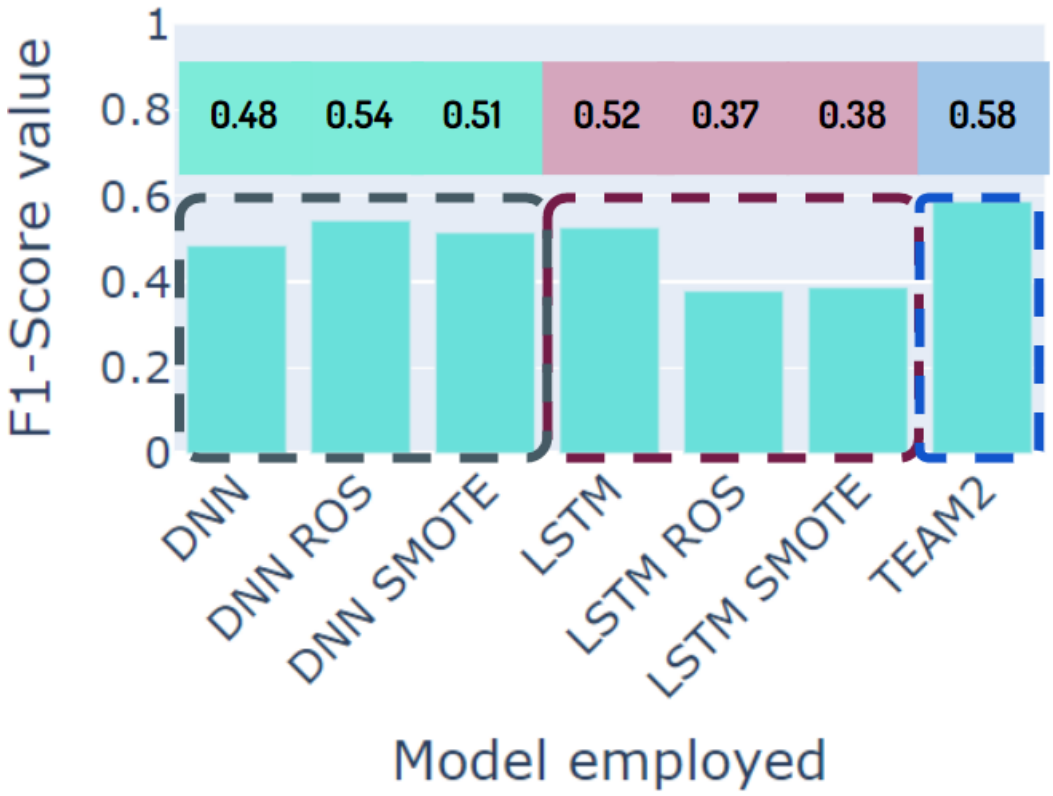


Figure 5.14. F1-score for methods employed in Federated Learning (FL) on the test set

than the TEAM2 and DNN ROS models. The last-mentioned means that applying oversampling techniques does not improve the result of the models in the LSTM approach. But, when using ROS over the augmented information, the performance of the FL model increases.

Considering the figure 5.15 results, the TEAM2 approach still generates the slowest procedure with a time of 78 minutes. In addition, the second less time-consuming approach ended up being the DNN ROS with 32 minutes, where using oversampling techniques makes the execution time increase [24]. Compared to the CL approach, the TEAM2 FL method has a faster execution with similar performance. On the opposite, DNN ROS and SMOTE ran slower but with a worse performance than their CL versions.

With FL, it is possible to control the model's performance in each local node. The latter is relevant to understanding whether some node is underperforming compared to others.

Figure 5.16 depicts the behaviour of each local node concerning the accuracy for both the training and validation datasets. In general, the models among the clients have similar behaviour, getting stable along the epochs. It is essential to clarify that the accuracy for train and validation seems close to each other, meaning that the

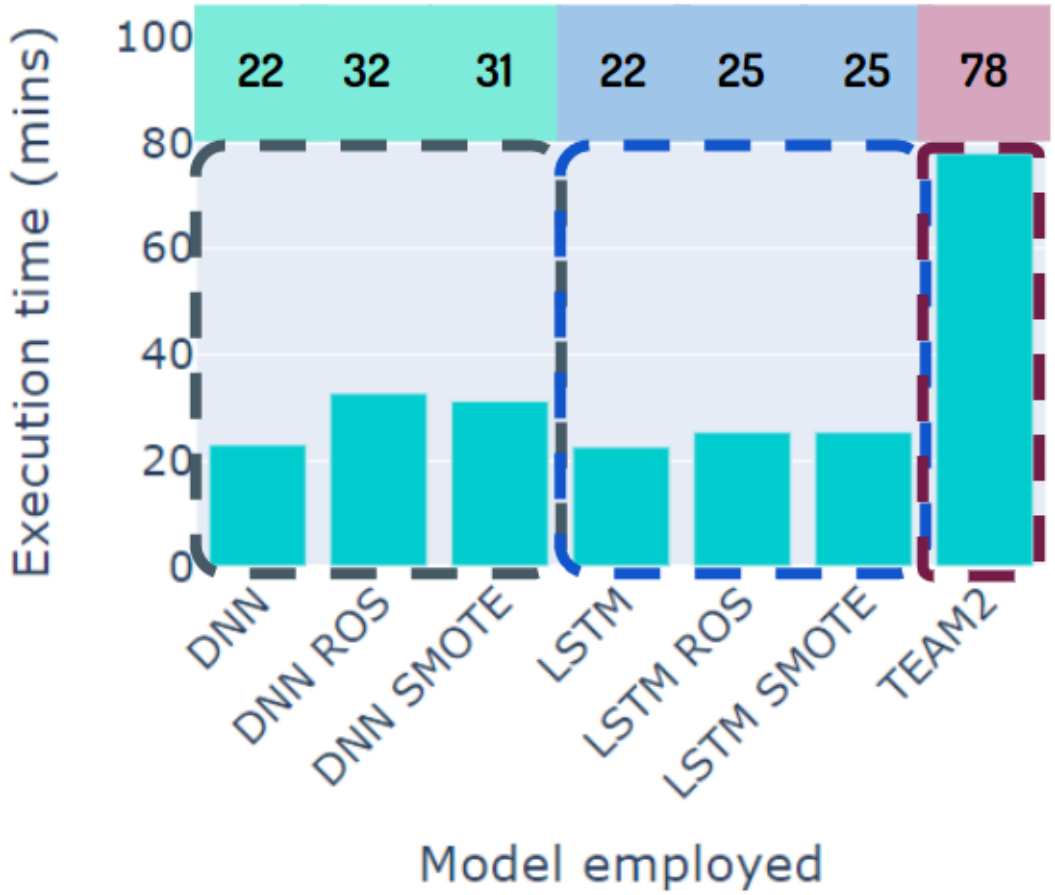


Figure 5.15. Execution times for the methods used in FL IID approach

models are getting robust results regarding overfitting.

In a a FL environment there is a concept called **communication round** (comm round). The latter begins when a model is trained inside each one of the local nodes. Later the weights of the models are passed to the global node to be aggregated there. And finally, the communication round finishes when each local model is updated with the new weights. Then, it is expected that in each comm round the performance of the model increases. Moreover, it should get stable after some trials.

Figure 5.17 establishes the behaviour of each metric for each communication round. As depicted, the performance gets stable after the 8th comm round approximately. Notice that, in the first comm round, all the metrics start low, but after some updates, the measurements get steady.

Table 5.3 depicts the metrics obtained over the test dataset and the execution time of the training phase. The latter also includes the performance of the second team of the 2020 Physionet competence and the DNN/LSTM models used as inspiration to construct all the approaches exposed in this work. The centralized TEAM2 model

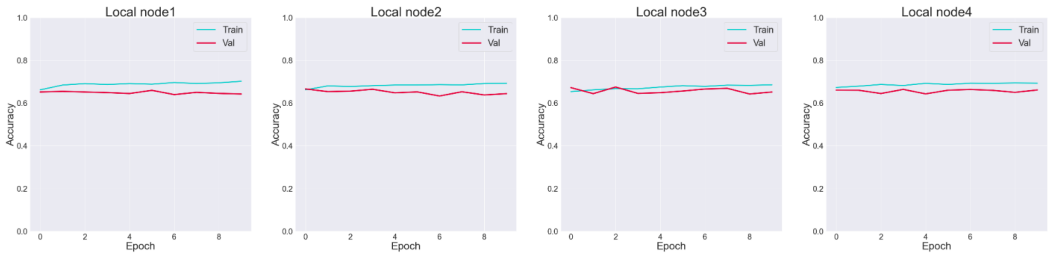


Figure 5.16. Train and Validation accuracy among local nodes for TEAM2 (the best model)

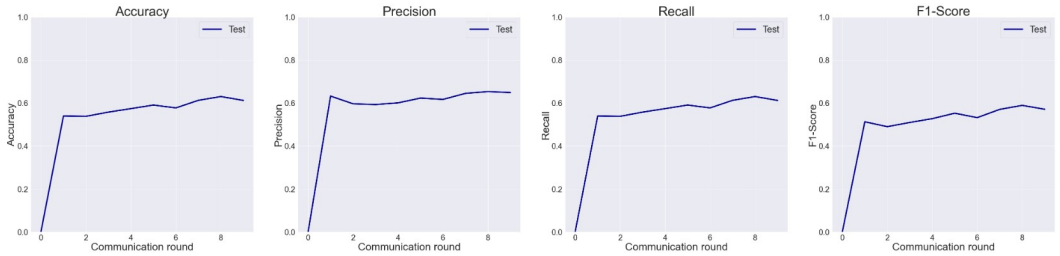


Figure 5.17. Metrics along communication rounds for TEAM2 (the best model) on the test set

outperforms all the proposals by at least two percentage points. Concerning the FL architecture, the TEAM2 FL is close enough to the centralized TEAM2 model, showing that the FL applied over an IID set has good behaviour.

5.3.2 Non-IID approach

This method was based on the idea of taking four random samples with repetition from the entire dataset of 41,894 registrations (local nodes). The given data in each local node (or client) will be Non-IID 4.2.4 approaches because to the unstratified random sampling. Because most ECGs shared across several devices are Non-IID, this scenario is far more plausible.

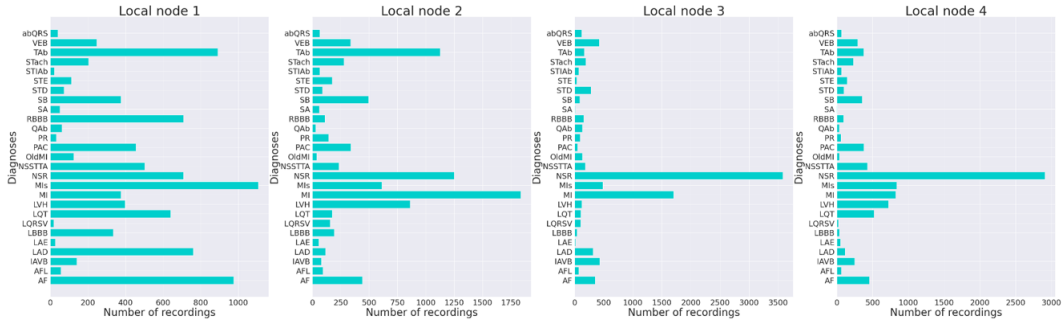


Figure 5.18. Label distribution for the filtered dataset by each local node for the Non-IID case

Method	CL					FL IID				
	Accuracy	Precision	Recall	F1-Score	Time	Accuracy	Precision	Recall	F1-Score	Time
Competence Team #2 [66]	**0.64**	0.64	**0.64**	**0.63**	122	**0.63**	**0.64**	**0.63**	**0.58**	78
Inspirational DNN [34]	0.50	0.46	0.50	0.47	**88**	-	-	-	-	-
Inspirational LSTM [34]	0.50	0.45	0.50	0.46	89	-	-	-	-	-
DNN	0.50	0.47	0.50	0.47	89	0.46	0.53	0.45	0.49	**22**
DNN ROS	0.61	**0.66**	0.61	0.61	90	0.55	0.59	0.54	0.55	32
DNN SMOTE	0.60	0.64	0.60	0.60	91	0.52	0.58	0.52	0.51	31
LSTM	0.51	0.47	0.51	0.39	91	0.48	0.58	0.48	0.52	**22**
LSTM ROS	0.38	0.51	0.38	0.39	91	0.39	0.48	0.38	0.38	25
LSTM SMOTE	0.39	0.49	0.39	0.39	89	0.39	0.47	0.39	0.38	25

Table 5.3. Metrics for Centralized (CL) and IID Federated Learning (FL -IID) in the test data set. Also included execution time for preprocessing and training in minutes.

The distribution among the four local nodes appears Non-IID, as shown in figure 5.18. The latter implies that diagnosis along the devices will differ. Furthermore, each local node has around 9,426 records. The ROS and SMOTE datasets behave similarly when divided into four clients, as demonstrated in figure 5.13 Naturally, all of the diagnoses in this scenario have nearly equal participation throughout the nodes, making them IID and balanced.

The modelling phase was done again with the four NON-IID datasets extracted. The 12-channel ECG arrhythmias were then classified using the TEAM2, DNN and LSTM techniques following an FL paradigm.

Figure 5.19. F1-score for methods employed in Federated Learning (FL) Non-IID case on the test set

We employed a centralized learning strategy (CL) to compare the NON-IID FL and CL implementations. The CL method implied appending the four created local nodes and utilizing that data to train a model. Then, ideally, the other algorithms should be as near to the CL approach as possible. The best results emerged when using the TEAM2 with the original datasets, as shown in figure 5.19, while the DNN

ROS and DNN LSTM techniques also performed well in this scenario, receiving an F1-score of 0.61 on the test set. On the other hand, the highest performance for LSTM was with the original data (LSTM), yet it was worse than the DNN ROS model. The previous suggests that using oversampling approaches does not improve the results of the models in the LSTM approach in this circumstance. However, using the TEAM2 solution made the FL model's performance the most useful of all the strategies and DNN on the ROS dataset is a good alternative due to its performance.

Figure 5.20. Execution times for the methods used in FL Non-IID approach

It is possible to obtain the execution times for the algorithms by analyzing picture 5.20. The TEAM2 method was the slowest, clocking in at 74 minutes. Compared to the CL method, the TEAM2 methodology with FL is faster and produces equivalent results. The same thing happens with DNN ROS and SMOTE. However, while LSTM performs worse than TEAM2 and DNN, it is significantly quicker in producing the results.

Along with the experimentation, we implemented a performance evaluation of the models by changing the number of local nodes diverse to 4. Then, there were some simulated scenarios by changing the number of clients from 2 to 10. Per each client, we trained a centralized model (CL) to determine how well the FL training was fitting regarding that CL reference point. It is relevant to clarify that the experiment was conducted for the IID and Non-IID approaches. Nevertheless, for the sake of the extension of this document, only the Non-IID result is reported because the performances of both methods were quite similar.

Figure 5.21 represents the F1-score measured in the test dataset for all the methods used by changing the number of local nodes. TEAM2 CL, DNN CL and LSTM CL are the reference model trained with all the data appended to a single

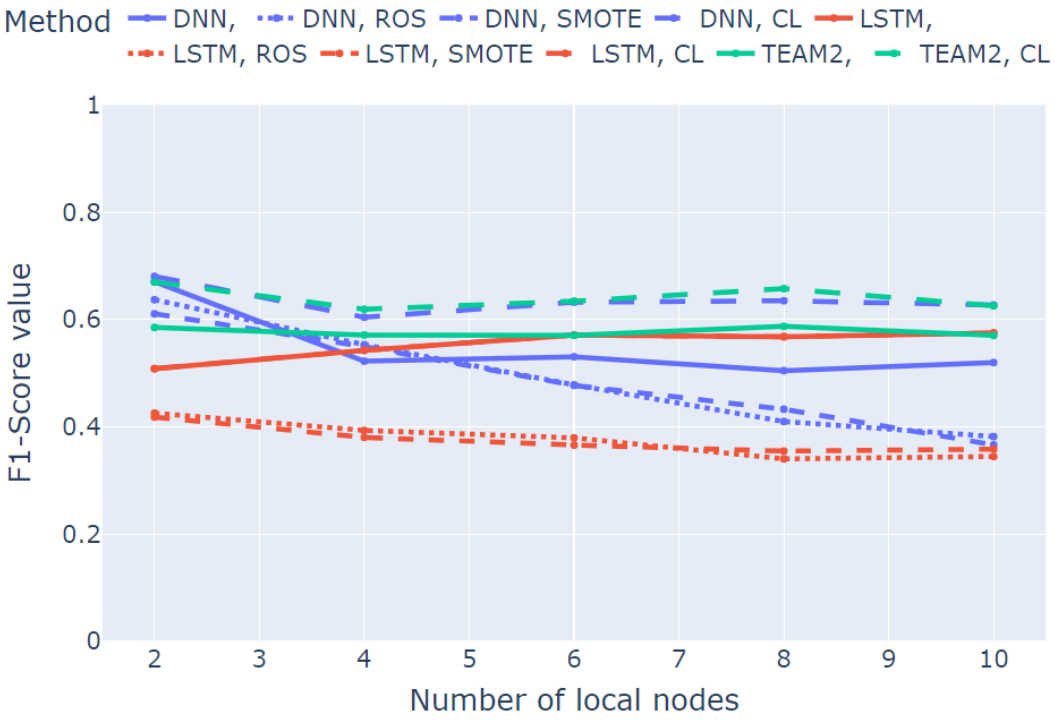


Figure 5.21. F1-score changing number of local nodes with FedAvg sing test set

dataset (centralized or standard learning). Then, we can see that when considering two local nodes, the best solution is DNN ROS since it is close enough to its reference model. On the other hand, when using four or more clients best solution is TEAM2, keeping more or less constant when increasing the number of clients. Nevertheless, when considering six or more nodes, LSTM on the original data is a good option since it is close to its CL performance and similar to the TEAM2 metric.

The execution time comparison changing the number of local nodes is demonstrated in figure 5.22. Typically, the higher the local nodes quantity, the faster the algorithms run. Moreover, all the techniques drastically decreased the running time when increasing the number of clients, but the performance also decreased. In the case of LSTM, it had one of the highest metrics with the fastest running time when using more clients.

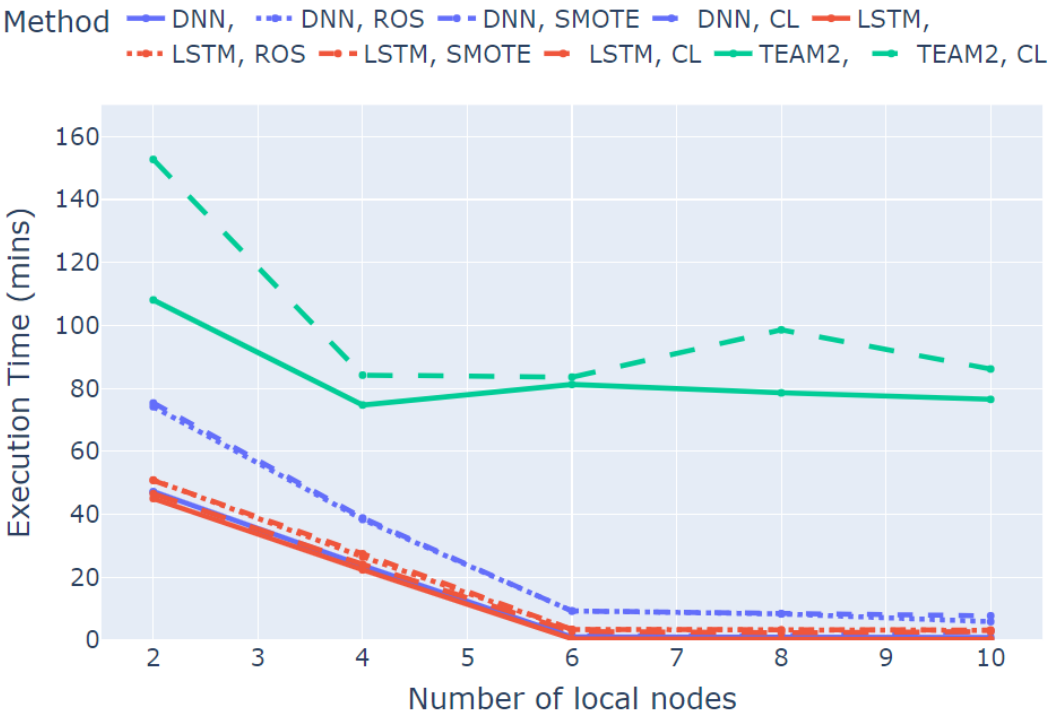


Figure 5.22. Running time changing number of local nodes with FedAvg

Chapter 6

Conclusions

6.1 Summary

In the end, the categorization of 12-channel ECG arrhythmias was implemented by using oversampling (and undersampling) approaches. Random Oversampling (ROS), in particular, performed well. The latter showed a considerable increment compared to the model trained with the original data. Although SMOTE performed well in a variety of cases, TEAM2 was chosen as the best strategy due to its prediction power.

Along with the experimentation, the solution from the second competence team provided remarkable results. Moreover, using deep neural networks over augmented datasets (ROS) also produced similar results. TEAM2 and DNN ROS attained equivalent behaviours compared to their respective CL approaches using few clients. Finally, using LSTM for larger clients demonstrated steady and analogous behaviour to the obtained in a CL, comparable to that obtained by the TEAM2.

6.2 Future Developments

Regarding future work, the FL architecture could use a different partition for the data. For example, each database can be employed as a local node. Thus, the six nodes will have a Non-IID distribution of arrhythmias. Another aspect to check is an alternative method to deal with the Non-IID property of the data. Fedprox, SCAFFOLD or FedNova [30] are proper candidates that could be carried out. Another venue for future investigations may include using Catboost and XGBoost in a federated learning architecture [29, 60] to check if they perform better for the 12-leads ECG arrhythmia classification.

Bibliography

[1] Arnold, D. and Wilson, T. What doctor? why ai and robotics will define new health. In *PwC* (2017).

[2] Basheer, I. and Hajmeer, M. Artificial neural networks: Fundamentals, computing, design, and application. *Journal of microbiological methods*, **43** (2001), 3. doi:10.1016/S0167-7012(00)00201-3.

[3] Bentéjac, C., Csörgő, A., and Martínez-Muñoz, G. A comparative analysis of xgboost (2019).

[4] Bogdanova, A., Attoh-Okine, N., and Sakurai, T. Risk and advantages of federated learning for health care data collaboration. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, **6** (2020), 04020031. doi:10.1061/AJRUA6.0001078.

[5] Borra, D., Andaló, A., Severi, S., and Corsi, C. On the application of convolutional neural networks for 12-lead ecg multi-label classification using datasets from multiple centers. In *2020 Computing in Cardiology*, pp. 1–4 (2020). doi:10.22489/CinC.2020.349.

[6] Canbek, G., Sagiroglu, S., Taskaya Temizel, T., and Baykal, N. Binary classification performance measures/metrics: A comprehensive visualized roadmap to gain new insights. pp. 821–826 (2017). doi:10.1109/UBMK.2017.8093539.

[7] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, **16** (2002), 321–357. Available from: http://dx.doi.org/10.1613/jair.953, doi:10.1613/jair.953.

[8] Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. In *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 785–794 (2016). doi:https://doi.org/10.1145/2939672.2939785.

[9] Darmawahyuni, A., Nurmaini, S., Sukemi, Caesarendra, W., Bhayyu, V., Rachmatullah, M. N., and Firdaus. Deep learning with a recurrent network structure in the sequence modeling of imbalanced data for ecg-rhythm classifier. *Algorithms*, **12** (2019). Available from: https://www.mdpi.com/1999-4893/12/6/118, doi:10.3390/a12060118.

[10] Demonbreun, A. and Mirsky, G. M. Automated classification of electrocardiograms using wavelet analysis and deep learning. In *2020 Computing in Cardiology*, pp. 1–4 (2020). doi:10.22489/CinC.2020.138.

[11] Divi, S., Lin, Y.-S., Farrukh, H., and Celik, Z. B. New metrics to evaluate the performance and fairness of personalized federated learning. (2021). Available from: https://arxiv.org/abs/2107.13173, doi:10.48550/ARXIV.2107.13173.

[12] Elayan, H., Aloqaily, M., and Guizani, M. Deep federated learning for iot-based decentralized healthcare systems. In *2021 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 105–109 (2021). doi:10.1109/IWCMC51323.2021.9498820.

[13] Fernández, A., Garcia, S., Herrera, F., and Chawla, N. Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. *Journal of Artificial Intelligence Research*, **61** (2018), 863. doi:10.1613/jair.1.11192.

[14] Gallo, C. *Artificial Neural Networks: tutorial* (2015). ISBN 9781466658882.

[15] Goldberger, A., et al. Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. *PhysioNet*, **101** (2000).

[16] Goldsworthy, S., Kleinpell, R., and Williams, G. *International Best Practices in Critical Care*. World Federation of Critical Care Nurses„ New York, NY : (2017.).

[17] Hanley, T. and Winter, L. What is a systematic review? *Counselling Psychology Review*, **28** (2013), 3.

[18] He, Z., Zhang, P., Xu, L., Bai, Z., Zhang, H., Li, W., Xia, P., and Chen, X. A novel convolutional neural network for arrhythmia detection from 12-lead electrocardiograms. In *2020 Computing in Cardiology*, pp. 1–4 (2020). doi:10.22489/CinC.2020.196.

[19] Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, **9** (1997), 1735. doi:10.1162/neco.1997.9.8.1735.

[20] Hossin, M. and M.N, S. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining and Knowledge Management Process*, **5** (2015), 01. doi:10.5121/ijdkp.2015.5201.

[21] Iaizzo, P. *Handbook of cardiac anatomy, physiology, and devices, third edition.* Springer International Publishing (2015). ISBN 9783319194639. doi:10.1007/978-3-319-19464-6.

[22] Ibraimi, L., Selimi, M., and Freitag, F. Bepoch: Improving federated learning performance in resource-constrained computing devices. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6 (2021). doi:10.1109/GLOBECOM46510.2021.9685095.

[23] Ibraimi, L., Selimi, M., and Freitag, F. Bepoch: Improving federated learning performance in resource-constrained computing devices. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6 (2021). doi:10.1109/GLOBECOM46510.2021.9685095.

[24] Jamali-Rad, H., Abdizadeh, M., and Singh, A. Federated learning with taskonomy for non-iid data. arXiv (2021). Available from: https://arxiv.org/abs/2103.15947, doi:10.48550/ARXIV.2103.15947.

[25] Jiang, M., Gu, J., Li, Y., Wei, B., Zhang, J., Wang, Z., and Xia, L. Hadln: Hybrid attention-based deep learning network for automated arrhythmia classification. *Frontiers in Physiology*, **12** (2021). Available from: https://www.frontiersin.org/article/10.3389/fphys.2021.683025, doi:10.3389/fphys.2021.683025.

[26] Kahale, L. A., Elkhoury, R., El Mikati, I., Pardo-Hernandez, H., Khamis, A., Schünemann, H., Haddaway, N., and Akl, E. Prisma flow diagrams for living systematic reviews: a methodological survey and a proposal. *F1000Research*, **10** (2021), 192. doi:10.12688/f1000research.51723.1.

[27] Kaushik, A. R. Types of Federated Learning (Retrieved on May 30, 2022). Available from: https://medium.com/@arjun.r.kaushik/types-of-federated-learning-1c0ce84fe7d5.

[28] Kopparapu, K., Lin, E., and Zhao, J. Fedcd: Improving performance in non-iid federated learning. (2020). Available from: https://arxiv.org/abs/2006.09637, doi:10.48550/ARXIV.2006.09637.

[29] Le, N. K., Liu, Y., Nguyen, Q. M., Liu, Q., Liu, F., Cai, Q., and Hirche, S. Fedxgboost: Privacy-preserving xgboost for federated learning (2021). Available from: https://arxiv.org/abs/2106.10662, doi:10.48550/ARXIV.2106.10662.

[30] Li, Q., Diao, Y., Chen, Q., and He, B. Federated learning on non-iid data silos: An experimental study. (2021). Available from: https://arxiv.org/abs/2102.02079, doi:10.48550/ARXIV.2102.02079.

[31] Luo, M., Chen, F., Hu, D., Zhang, Y., Liang, J., and Feng, J. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. (2021). Available from: https://arxiv.org/abs/2106.05001, doi:10.48550/ARXIV.2106.05001.

[32] Luo, X., Yang, L., Cai, H., Tang, R., Chen, Y., and Li, W. Multi-classification of arrhythmias using a hcrnet on imbalanced ecg datasets. *Computer Methods and Programs in Biomedicine*, **208** (2021), 106258. Available from: https://www.sciencedirect.com/science/article/pii/S0169260721003321, doi:https://doi.org/10.1016/j.cmpb.2021.106258.

[33] Murat, F., Yildirim, O., Talo, M., Baloglu, U., Demir, Y., and Acharya, U. Application of deep learning techniques for heartbeats detection using ecg signals analysis and review. *Computers in Biology and Medicine*, **120** (2020). doi:10.1016/j.compbiomed.2020.103726.

[34] Murat, F., Yildirim, O., Talo, M., Baloglu, U. B., Demir, Y., and Acharya, U. R. Application of deep learning techniques for heartbeats detection using ecg signals-analysis and review. *Computers in Biology and Medicine*, **120** (2020), 103726. Available from: https://www.sciencedirect.com/science/article/pii/S0010482520301104, doi:https://doi.org/10.1016/j.compbiomed.2020.103726.

[35] Nguyen, Q., Ly, H.-B., Lanh, H., Al-Ansari, N., Le, H., Van Quan, T., Prakash, I., and Pham, B. Influence of data splitting on performance of machine learning models in prediction of shear strength of soil. *Mathematical Problems in Engineering*, **2021** (2021). doi:10.1155/2021/4832864.

[36] Organization, W. H. *Global diffusion of eHealth: making universal health coverage achievable: report of the third global survey on eHealth*. World Health Organization (2017).

[37] Perez, E. and matthewreyna. physionetchallenges (2021). Available from: https://github.com/physionetchallenges/physionetchallenges.github.io/blob/master/2020/Dx_map.csv.

[38] Perez Alday, E., et al. Classification of 12-lead ecgs: the physionet/computing in cardiology challenge 2020. (2020). doi:10.1101/2020.08.11.20172601.

[39] PhysioNet. Classification of 12-lead ecgs: The physionet/computing in cardiology challenge 2020 (Retrieved on June 01, 2022). Available from: https://www.physionet.org/content/challenge-2020/1.0.1/.

[40] PhysioNet. Moody challenge overview (Retrieved on May 31, 2022). Available from: https://physionet.org/about/challenge/moody-challenge-overview.

[41] Popescu, M.-C., Balas, V., Perescu-Popescu, L., and Mastorakis, N. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, **8** (2009).

[42] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. Catboost: unbiased boosting with categorical features. arXiv (2017). Available from: https://arxiv.org/abs/1706.09516, doi:10.48550/ARXIV.1706.09516.

[43] Rath, A., Mishra, D., Panda, G., and Satapathy, S. C. Heart disease detection using deep learning methods from imbalanced ecg samples. *Biomedical Signal Processing and Control*, **68** (2021), 102820. Available from: https://www.sciencedirect.com/science/article/pii/S1746809421004171, doi:https://doi.org/10.1016/j.bspc.2021.102820.

[44] Rattenbury, T., Hellerstein, J. M., Heer, J., Kandel, S., and Carreras, C. *Principles of Data Wrangling: Practical Techniques for Data Preparation*. O'Reilly Media, Inc., 1st edn. (2017). ISBN 1491938927.

[45] Raza, A., Tran, K. P., Koehl, L., and Li, S. Designing ecg monitoring healthcare system with federated transfer learning and explainable ai. *Knowledge-Based Systems*, **236** (2022), 107763. Available from: https://www.sciencedirect.com/science/article/pii/S0950705121009862, doi:https://doi.org/10.1016/j.knosys.2021.107763.

[46] Rieke, N., et al. The future of digital health with federated learning. *NPJ digital medicine*, **3** (2020), 1.

[47] Sakib, S., Fouda, M. M., Md Fadlullah, Z., Abualsaud, K., Yaacoub, E., and Guizani, M. Asynchronous federated learning-based ecg analysis for arrhythmia detection. In *2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, pp. 277–282 (2021). doi:10.1109/MeditCom49071.2021.9647636.

[48] Sampson, M. and Mcgrath, A. Understanding the ecg. part 1: Anatomy and physiology. *British Journal of Cardiac Nursing*, **10** (2015), 548. doi:10.12968/bjca.2015.10.11.548.

[49] Sampson, M. and Mcgrath, A. Understanding the ecg part 2: Ecg basics. *British Journal of Cardiac Nursing*, **10** (2015), 588. doi:10.12968/bjca.2015.10.12.588.

[50] Sannino, G. and De Pietro, G. A deep learning approach for ecg-based heartbeat classification for arrhythmia detection. *Future Generation Computer Systems*, **86** (2018), 446. Available from: https://www.sciencedirect.com/science/article/pii/S0167739X17324548, doi:https://doi.org/10.1016/j.future.2018.03.057.

[51] Savalia, S. and Emamian, V. Cardiac arrhythmia classification by multilayer perceptron and convolution neural networks. *Bioengineering*, **5** (2018). Available from: https://www.mdpi.com/2306-5354/5/2/35, doi:10.3390/bioengineering5020035.

[52] Stefanski, R. Data wrangling in 6 steps: An analyst's guide for creating useful data (Retrieved on May 05, 2022). Available from: https://hevodata.com/learn/data-wrangling/.

[53] Surda, J., Lovas, S., Pucik, J., and Jus, M. Spectral properties of ecg signal. pp. 1–5 (2007). ISBN 1-4244-0821-0. doi:10.1109/RADIOELEK.2007.371653.

[54] Talebi, S. The wavelet transform (2020). Available from: https://towardsdatascience.com/the-wavelet-transform-e9cfa85d7b34.

[55] VG, S. and K.P., S. Towards identifying most important leads for ecg classification. a data driven approach employing deep learning. *Procedia Computer Science*, **171** (2020), 602. Third International Conference on Computing and Network Communications (CoCoNet'19). Available from: https://www.sciencedirect.com/science/article/pii/S1877050920310322, doi:https://doi.org/10.1016/j.procs.2020.04.065.

[56] Vishwa, A., Lal, M., Dixit, S., and Varadwaj, P. Clasification of arrhythmic ecg data using machine learning techniques. *International Journal of Interactive Multimedia and Artificial Intelligence*, **1** (2011), 67. doi:10.9781/ijimai.2011.1411.

[57] Walraven, G. *Basic Arrhythmias*. Pearson Education (2014). ISBN 9780133763577. Available from: https://books.google.ne/books?id=5f67AQAAQBAJ.

[58] Wang, F., Casalino, L. P., and Khullar, D. Deep learning in medicine—promise, progress, and challenges. *JAMA internal medicine*, **179** (2019), 293.

[59] Wang, L., Xu, S., Wang, X., and Zhu, Q. Addressing class imbalance in federated learning. (2020). Available from: https://arxiv.org/abs/2008.06217, doi:10.48550/ARXIV.2008.06217.

[60] Yang, Q., Fan, L., and Yu, H. *Federated Learning: Privacy and Incentive*, vol. 12500. Springer Nature (2020).

[61] Yang, Q., Liu, Y., Chen, T., and Tong, Y. Federated machine learning: Concept and applications. (2019). Available from: https://arxiv.org/abs/1902.04885, doi:10.48550/ARXIV.1902.04885.

[62] Yeganeh, Y., Farshad, A., Navab, N., and Albarqouni, S. Inverse distance aggregation for federated learning with non-iid data. (2020). Available from: https://arxiv.org/abs/2008.07665, doi:10.48550/ARXIV.2008.07665.

[63] Zhang, D. Interpretable deep learning for automatic diagnosis of 12-lead electrocardiogram (2021). Available from: https://github.com/onlyzdd/ecg-diagnosis.

[64] Zhang, M., Wang, Y., and Luo, T. Federated learning for arrhythmia detection of non-iid ecg. In *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, pp. 1176–1180 (2020). doi:10.1109/ICCC51575.2020.9344971.

[65] Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., and Chandra, V. Federated learning with non-iid data. (2018). Available from: https://arxiv.org/abs/1806.00582, doi:10.48550/ARXIV.1806.00582.

[66] Zhao, Z., Fang, H., Relton, S. D., Yan, R., Liu, Y., Li, Z., Qin, J., and Wong, D. C. Adaptive lead weighted resnet trained with different duration signals for classifying 12-lead ecgs. In *2020 Computing in Cardiology*, pp. 1–4. IEEE (2020).